Universidad de Sevilla

Facultad de Matemáticas

Departamento de Estadística e Investigación Operativa

Introducción a la Inteligencia de Negocios. con ayuda de R.

Autora:
Mª Ángeles Mora Maqueda

Tutor:
Pedro Luis Luque Calvo

19 de Junio de 2017

Índice general

Abstract

In this work we will obtein a first completed vision of bussines intelligence. To achieve this aim, we are going to present the main aspects of bussines intelligence acrros four chapters of this book.

First of all, we have to understand what the concept of "bussiness intelligence" means and know the main elements that make this complex concept. We will try to answer the most important questions: *"What it is?"*, *"When and how shoul we use it?"*, *"how does it work?" "and why it is so importan and essential to the good development of a company"*.

On the Second chapter, we are going to introduce the main commercials software that exist around bussiness intelligence. Specially, we will talk about Qlik, Tableau, Pentahoo and BI Microsoft. We will know their principal characteristics to can comparate each one through their pros and cons.

On the next Chapter we are going to present R as a valid option to be a software for bussiness intelligence. We will go across the main librarys that will allow us to realize the steps bussiness intelligence needs.

Finally, we use the knowledge of previous chapters to create an example dashboard with the technology proportionated by R.

Capítulo 1

Introducción a la inteligencia de negocios

1.1. Primer acercamiento a la inteligencia de negocios (BI)

Vivimos en un mundo donde la generación de nueva información es continua. Esto hace que el volumen de datos que se maneje en cualquier empresa, incluso en la más modesta, sea de una manipulación y visualización inabarcable si no se disponen de herramientas o aplicaciones que lo permitan. A raíz de esto, surge la **inteligencia de negocios** como un concepto orientado al tratamiento inmediato de los datos, la información y, en definitiva, el conocimiento, con el fin de mejorar los procesos de cualquier organización frente a las exigencias del mercado.

El término **business intelligence** surge de una tecnología de *soporte o ayuda a la decisión*.

En vistas a la realización de este trabajo, vamos a emplear el término **inteligencia de negocios**, el cual es la traducción más aceptada del inglés, aunque en algunas ocasiones también utilizaremos las siglas **BI**.

El potente crecimiento que ha experimentado **la inteligencia de negocios** se entiende, principalmente, a raíz de dos aspectos:

- Aumento en el volumen de datos.
- Aumento de la capacidad de los sistemas de almacenamiento.

La mayoría de compañías son altamente competitivas y las oportunidades de negocio son muy sensibles al momento en que se toman diferentes decisiones. Tomar decisiones acertadas, y rápidamente, mejora el rendimiento y genera una gran ventaja competitiva frente a otras compañías. Una información discontinua puede acarrear aspectos negativos al devenir de la empresa, como la pérdida de clientes o la continuación de la fabricación de un producto que realmente no interesa.

Un ejemplo clásico de la aplicación de la inteligencia de negocios es la correlación entre los pañales y la cerveza. En una de las grandes superficies más conocidas de EEUU, Wal Mart, se empezó a utilizar software analítico para combinar el análisis de los datos de compras con los demográficos obtenidos de la tarjeta de lealtad de los clientes. A partir de este análisis, se descubrieron una serie de asociaciones, algunas esperadas y lógicas (cereales y leche, ginebra con tónica y limones), y una completamente inesperada entre dos productos, que aparentemente poca relación parecen tener: pañales y cerveza. El descubrimiento de esta correlación hizo que se moviera la cerveza al lado de los pañales, y se obtuvo un incremento en las ventas.

Cada día las organizaciones están protagonizando cambios en sus negocios y actividades: nuevas áreas, cambios organizados, evoluciones tecnológicas, nuevos comportamientos de los clientes, ... Por eso, la información de gestión supone una pieza clave para el seguimiento y control del negocio, así como para el apoyo en la toma

de decisiones. Por ello, el alcance del éxito de toda empresa o negocio depende de la información y, más aún, del tratamiento y explotación que se haga de ella.

A partir de la gestión del conocimiento, surge el concepto de inteligencia de negocios, siendo este el conjunto de estrategias, acciones y herramientas enfocadas a la administración y creación de conocimiento mediante el análisis de datos existentes en una organización o empresa.

En este sentido, la inteligencia de negocios ofrece la posibilidad de obtener ventajas empresariales, mediante su aplicación en los procesos de toma de decisiones. El filósofo Albin Tofel manifiesta: *"El dueño de la información es el dueño del poder"*, y a esta afirmación hay que añadirle: en el lugar y momento oportuno.

Figura 1.1: Ejemplo de cuadro de mandos mediante mapas: Visualizar el dónde y el porqué

Además de la importancia que tiene para la toma de decisiones la inteligencia de negocios, también presenta una solución para conocer mejor a nuestra empresa a través de los datos, permitiendo observar qué está sucediendo, conocer el porqué ocurre, predecir qué ocurriría y analizar y decidir el camino a seguir (decisiones).

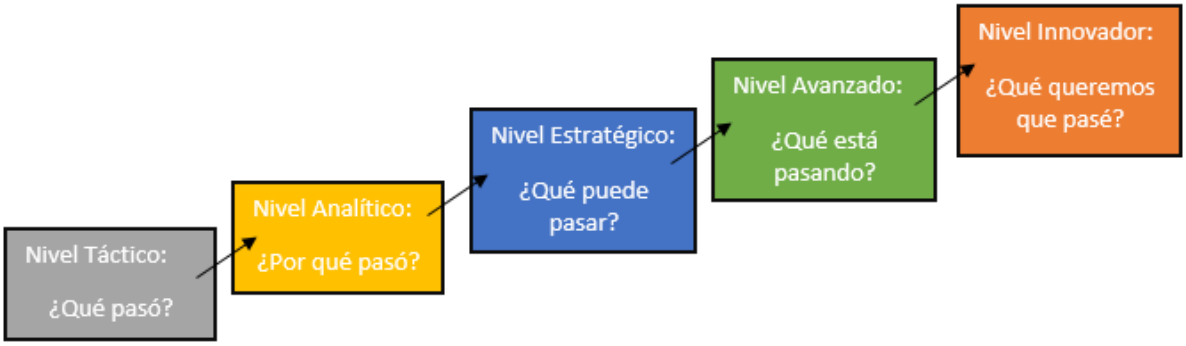

Figura 1.2: Preguntas que responde la inteligencia de negocios

Una vez motivada la importancia de la inteligencia de negocios, nos preguntamos qué debemos hacer para elegir bien. Para poder tomar decisiones es imprescindible disponer de una información adecuada que nos permita extraer de ella aquellos aspectos que son relevantes de una manera fiable, eficaz, sencilla y rápida.

Un ejemplo donde podemos ver como la inteligencia de negocios influye en una empresa es el siguiete:

Una franquicia de hoteles a nivel nacional utiliza aplicaciones de inteligencia de negocios para llevar un registro estadístico en el que recoge, por ejemplo, el porcentaje promedio de ocupación del hotel y los días de estancia de cada huésped. Con esta información ellos pueden:

- Calcular la rentabilidad de cada hotel en cada temporada del año.
- Determinar quién es su segmento de mercado.
- Calcular la participación de mercado de la franquicia y de cada hotel.
- Identificar oportunidades y amenazas.

Estas son sólo algunas de las formas en que una empresa u organización puede obtener beneficios por la implementación de programas de inteligencia de negocios. Otros aspectos los podemos ver en la siguiente imagen:

Figura 1.3: Algunos de los beneficios que se pueden obtener a través del uso del BI

1.1.1. Definición Inteligencia de Negocios

Una definición de *inteligencia de negocios* que abarque toda la tesitura de la misma es algo complicado. A continuación se recogen algunas de las definiciones que se han hecho de ella:

1. **CIO**: "Un conjunto de aplicaciones informáticas usadas para analizar datos en bruto". [1]
2. **Information Builders**: "Una amplia gama de soluciones informáticas que permiten a las compañías u organizaciones obtener el conocimiento necesario en sus operaciones a través de herramientas de análisis e informe".[2]
3. **Forrester**: "Un conjunto de metodologías, procesos, ingeniería informática y aplicaciones tecnológicas que permiten obtener información ventajosa a través de procesos de análisis, informes, gestión del rendimiento y entrega de información.[3]
4. **OLAP**: "Tecnologías, aplicaciones y prácticas para recolectar, integrar, analizar y presentar información del negocio".[4]
5. **Technopedia**: "El uso de aplicaciones informáticas con el fin de identificar, descubrir y analizar los datos relevantes para el negocio, como los ingresos por las ventas, los costos y los beneficios."[5]

Por lo tanto, podríamos construir una definición de la inteligencia de negocios considerando todos los aspectos y matices mencionados:

La **inteligencia de negocios** se define como la transformación de los datos en conocimiento para sustentar la toma de decisiones. Su aplicación en el ámbito estratégico y táctico, en el momento y lugar oportuno,

[1] http://www.cio.com/article/2439504/business-intelligence/business-intelligence-business-intelligence-definition-and-solutions.html.
[2] http://www.informationbuilders.com/business-intelligence
[3] https://www.forrester.com/Business-Intelligence
[4] http://olap.com/learn-bi-olap/olap-bi-definitions/business-intelligence/
[5] https://www.techopedia.com/definition/345/business-intelligence-bi

permite obtener una ventaja competitiva a través de la gestión de este conocimiento, orientándolo al apoyo a la toma de decisiones, con el propósito de incrementar la efectividad de la empresa.

1.1.2. Diferencia entre datos, información y conocimiento.

> "Una organización puede ser rica en datos y pobre en información, sino sabe como identificar, resumir y categorizar los datos" (Madnick, 1993)

Para entender el importante papel y dónde entra en juego **la inteligencia de negocios** vamos a definir los siguientes tres conceptos:

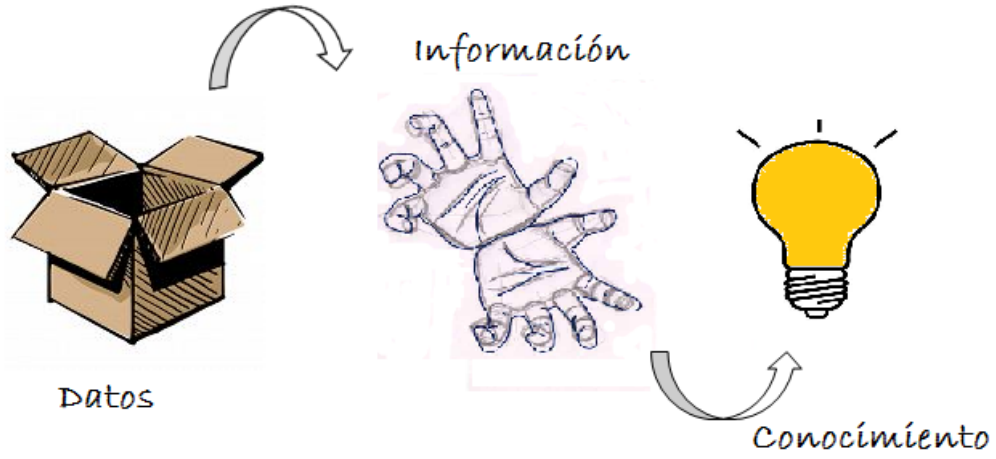

Figura 1.4: Dato-Información-Conocimiento

1. **Datos**: Es conocido que las empresas almacenan una gran cantidad de datos. Sin embargo, el dato es una simple transacción, que sin ningún tratamiento o manipulación no aporta absolutamente nada. A pesar de esto, son la base del funcionamiento de cualquier empresa ya que a partir de la información que surge de ellos se tomarán las decisiones.
2. **Información**: El dato pasa a ser información cuando se transforma añadiéndole significado y valor.

   Se pueden distinguir tres tipos de información:

   - *Información Técnico Operativa*: Es la información requerida por el personal operativo para realizar las operaciones que mantienen en funcionamiento el negocio.
   - *Información Táctica*: Es la información usada por coordinadores de área y directores operativos, para dirigir la ejecución de las operaciones por parte del personal operativo.
   - *Información Estratégica*: Es la información usada por los altos directivos para direccionar el negocio hacia la consecución de los objetivos gerenciales.
3. **Conocimiento**: El conocimiento no es un concepto simple. Este se deriva del análisis de la información. Es útil para incorporar nuevas directrices de comportamiento o estrategias a la empresa. El conocimiento nace a partir de varios elementos como la experiencia del usuario, valores e información que nos permite entender y juzgar la información.

En este último aspecto es donde ha aportado más la inteligencia de negocios. Cuando tenemos conocimiento del negocio a partir de la información obtenida es el momento en el que es posible plantear estrategias y analizar las fortalezas y debilidades de éste.

La información y el conocimiento que se genera en la organización se consume en diferentes momentos según el nivel:

Plazo	Nivel	Uso
Corto Plazo	Operacional y Administrativo	Obtención y control de datos
Medio Plazo	De Conocimientos	Decisiones tácticas
Largo Plazo	Estratégico	Decisiones estratégicas

- Por un lado, la información operativa y táctica es provista por los sistemas transaccionales.
- Sin embargo, la información estratégica asociada al el conocimiento, la toma de decisiones y el direccionamiento hacia los objetivos del negocio, debe ser provista por los sistemas de Inteligencia de Negocios.

1.2. Evolución histórica

Para tener una visión panorámica sobre como ha ido evolucionando la inteligencia de negocios vamos a mostrar una breve y esquemática cronología, resumiendo la información aportada por *"The Definitive Guide of Bussines Intelligence, de Betterbuys.com"*.

- 1988: Aparece el término "Bussines Intelligence" definido por Howard Dresne como lo conocemos hoy.
- 1990: Empiezan a desarrollarse las consultas y los informes a través de técnicas informáticas.
- 1997: Comienza a utilizarse más ampliamente el término "Bussines Inteligence".
- 1999: La utilización del análisis para predecir empieza a cambiar la manera de operar de los negocios.
- 2001: Primera aparición del término SAAS "Software as a Service".
- 2005: El aumento de la presencia de los medios de comunicación genera un crecimiento exponencial de los datos.
- 2010: La utilización de la inteligencia de negocios se incrementa en un 35 % respecto al año anterior.
- 2014: El mercado de la inteligencia de negocios alcanza 81 mil millones de dólares.
- 2018: Se espera que se produzca un importante aumento en el mercado, alcanzando unos 136 mil millones de dólares.

1.3. Objetivos y Razones para la Inteligencia de Negocios

El objetivo principal de una empresa es ser competitivo y tomar las decisiones adecuadas en el momento justo. Estas decisiones pueden ser sobre cualquier aspecto del día a día del negocio: cómo incrementar la eficiencia en las campañas de marketing, decidir dónde y cuándo abrir nuevos mercados, o llegar a conocer mejor las necesidades de los clientes con el fin de mejorar los productos y servicios ofrecidos. Podemos ver un ejemplo de **cuadro de mandos** que muestra información sobre los clientes en la figura 1.5

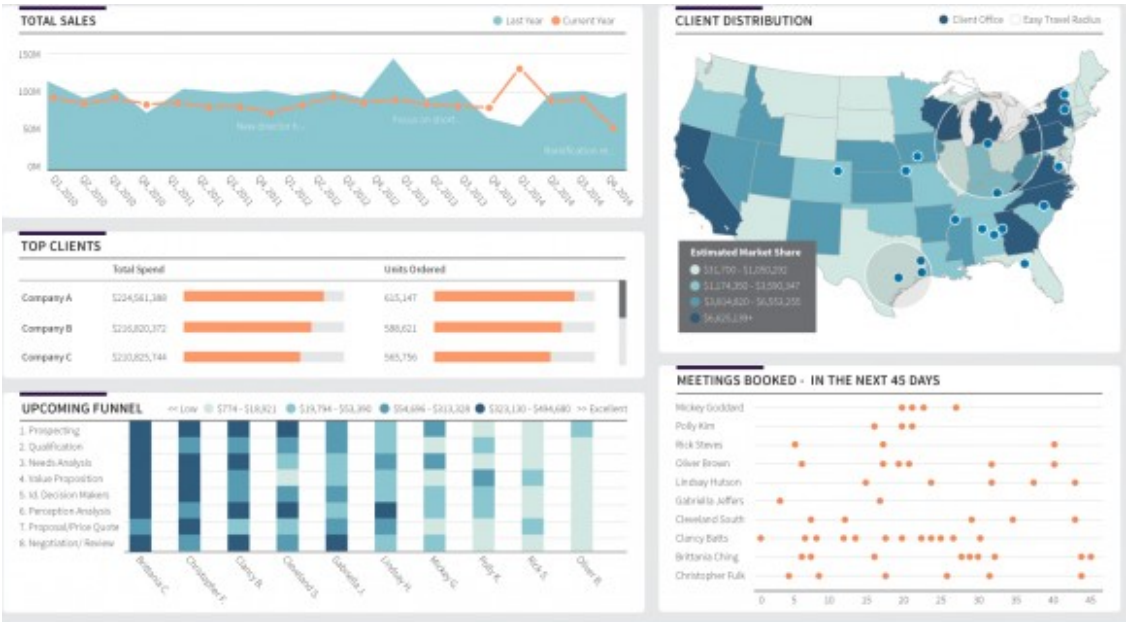

Figura 1.5: Cuadro de mandos, usado en inteligencia de negocios, sobre información de ventas, clientes,etc

A continuación, consideramos varios aspectos dónde la inteligencia de negocios puede hacer crecer a la empresa, conduciéndola hacia el éxito:

1. **Aumentar la productividad**. Al pensar en la gestión de una empresa, irremediablemente se nos viene a la mente la creación de largos y tediosos informes después de analizar gran cantidad de datos. Evidentemente, esto conlleva un consumo de tiempo que se puede ahorrar mediante la inteligencia de negocios. Con el uso de programas o aplicaciones se pueden pulir los datos y generar informes de manera mucho más automática, lo que permitiría optimizar el tiempo de los empleados en otras tareas.

2. **Incrementar la rentabilidad**. La inteligencia de negocios puede ayudar a detectar patrones y tendencias como, por ejemplo, la combinación de ventas de productos que generen mayor utilidad a la empresa.

3. **Seguimiento y mejora de la gestión de clientes**. Especialmente en negocios relacionados con las ventas sería interesante tener un seguimiento actualizado de los clientes. Esto es, tener un conjunto de datos sobre estos y darles sentido con el objetivo de mejorar el servicio que se les presta. Para ello es útil detectar patrones de comportamientos o necesidades de los clientes, ver la potencialidad de nuevos clientes, plantear estrategias sobre la post-venta. Es decir, en general, la inteligencia de negocios nos puede permitir visualizar tablas y gráficas que nos den información útil y ordenada para obtener conocimiento sobre los clientes y sus necesidades con el fin de ofrecerles un mejor servicio y retenerlos, así como vislumbrar posibles nuevas vías que acerquen a nuevos consumidores. Otro aspecto interesante es cómo detectar las principales razones por las que perdemos clientes y, en consecuencia, poder identificar los clientes con altas probabilidades de irse.

4. **Reducir costos**. La inteligencia de negocios es de gran utilidad para construir supuestos y posibles escenarios *"¿qué pasa sí...?"*, de tal manera que, por ejemplo, podemos obtener un análisis de costos muy detallado de cada área de nuestra empresa, así como optimizar la gestión de inventarios, por ejemplo disparando alertas que generen órdenes de compra.

5. **Cercanía a todas las áreas de la empresa**. Las empresas se organizan en secciones o áreas de manera que la acción y el trabajo global del negocio se divide. Esto hace que, inevitablemente, se analicen los datos y se obtenga información de manera parcelada, por lo que la visión global de la empresa se puede perder. Mediante la inteligencia de negocios podemos utilizar aplicaciones que permitan integrar

los datos de diferentes áreas o fuentes y paliar la falta de visibilidad en las operaciones de la empresa de diferentes áreas. Esto permite el acceso a toda la información relevante del negocio de manera rápida y eficaz.

6. **Reducir los riesgos**. En instituciones crediticias y de seguros la inteligencia de negocios es de gran ayuda, ya que analizando el historial de los clientes, puede detectar patrones que alerten sobre clientes con comportamientos fraudulentos o sobre el alto riesgo que puede conllevar a aceptar a un individuo como cliente.

7. **Acceso de la información sencilla**. El conjunto de usuarios que requiere del acceso a la información es amplio y se está incrementando. Además, muchos de esos usuarios no tienen conocimientos sobre análisis y tratamiento de los datos. La utilización de la inteligencia de negocios acerca, de manera entendible, la información ya tratada a todos los usuarios lo cual hará más óptimo su aprovechamiento y su trabajo. Este hecho también incrementa la colaboración y la interacción.

8. **Información actualizada**. La buena gestión de un negocio está íntimamente relacionada con tener información actual que permita tomar decisiones en base a la realidad. Los datos que llegan a la empresa pueden ser continuos, de manera que a través de una aplicación de inteligencia de negocios se permite automatizar su análisis, orden y organización de manera correcta. De esta forma podremos sacar información relevante e importante de ellos que nos permita obtener respuestas en un tiempo más que razonable.

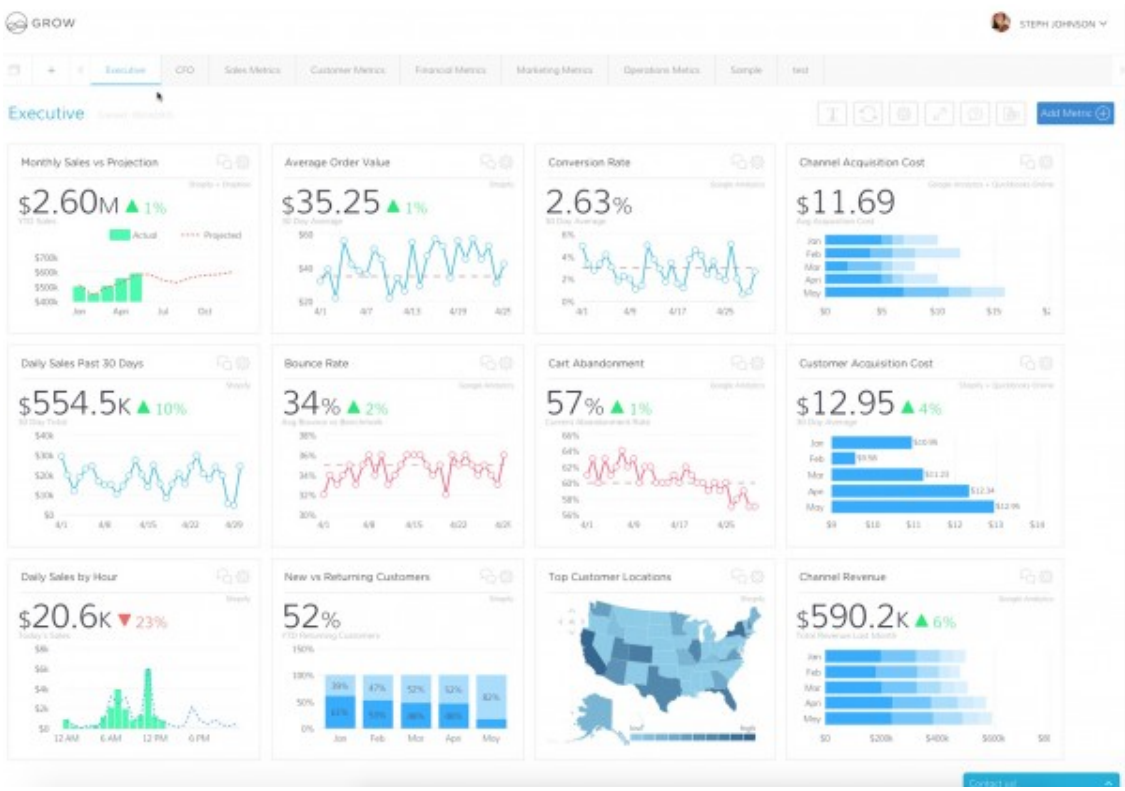

Figura 1.6: Cuadro de mando que muestra diferente información sobre una empresa dedicada a las ventas: Ventas mensuales VS proyección, ventas diarias de los últimos 30 días, ventas diarias por franjas horaria,zonas con más clientes,etc

1.4. Sectores donde se utiliza la Inteligencia de negocios.

La **inteligencia de negocios** puede dar servicio a diversos departamentos o sectores de las empresa:

1. **Sector Comercial**. Mejora el análisis y aprovechamiento de la información de los clientes así como el panorama de la competencia. Facilita la creación de campañas comerciales más eficientes y permite ofrecer un sistema de atención a los consumidores mejorado, más individualizado a las necesidades de cada uno.

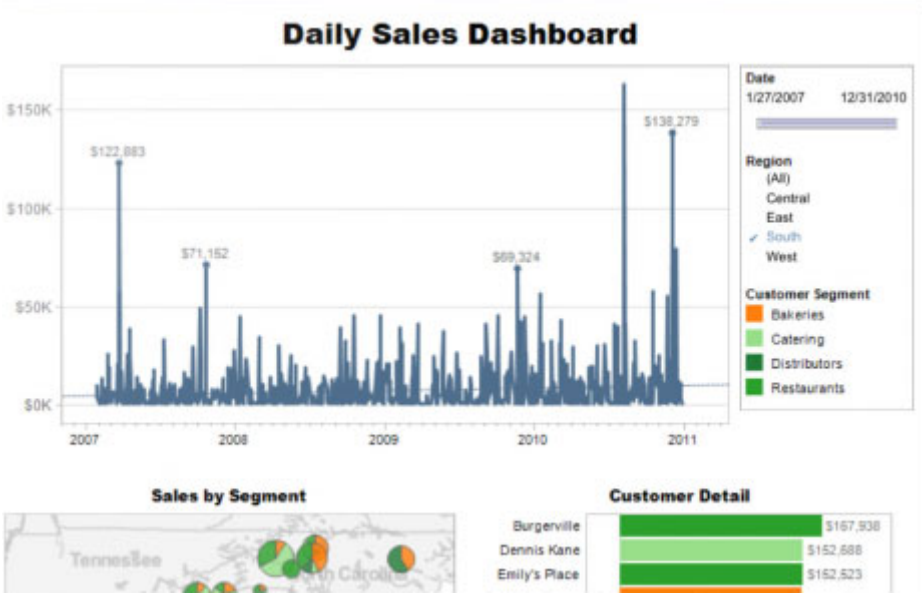

Figura 1.7: Cuadro de mando sobre ventas (para Sector Comercial)

2. **Áreas de control del Riesgo**. Facilita el análisis del riesgo: creación de modelos de previsión y estrategias de actuación ante posibles situaciones, estudio de solvencia frente a problemas futuros factibles, determinación del precio adecuado de los productos teniendo en cuenta cuánto estaría dispuesto a pagar un cliente por él, etc.

3. **Sectores de Gestión y Mando**. Permite tener cubierta toda la información interna del negocio analizándola de manera eficaz y presentándola mediante cuadros de mando de fácil visualización e interpretación para la alta dirección.

4. **Área de Recursos Humanos**. Detecta ante que aspectos los distintos departamentos pueden verse afectados negativamente, así como aquello que incentiva un clima de trabajo más propicio. Permite analizar la satisfacción de los empleados, el absentismo laboral, el beneficio por hora que genera cada empleado, etc.

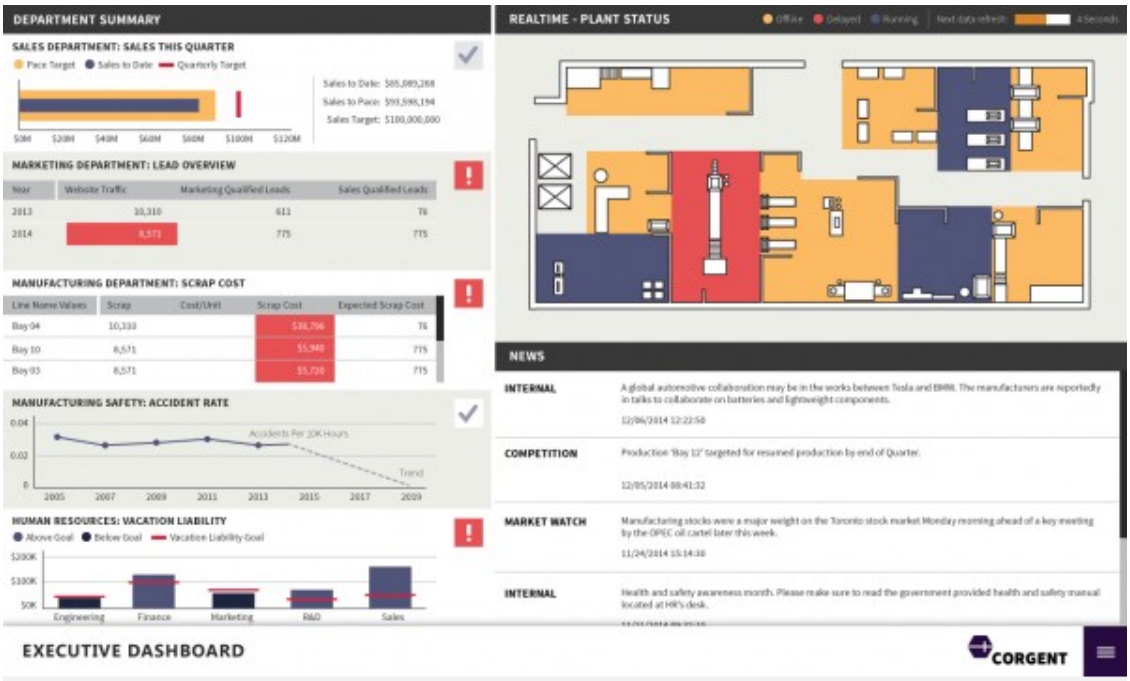

Figura 1.8: Cuadro de mandos que presenta información sobre la organización y actualidad de los diferentes departamentos de una empresa.

5. **Área de Ventas**. Análisis de ventas, detección de clientes importantes, análisis de productos, líneas y mercados, pronósticos y proyecciones.
6. **Área de Marketing**. Segmentación y análisis de clientes, seguimiento a nuevos productos. Podemos ver un ejemplo de cuadro de mandos dedicado a este fin en la figura 1.9

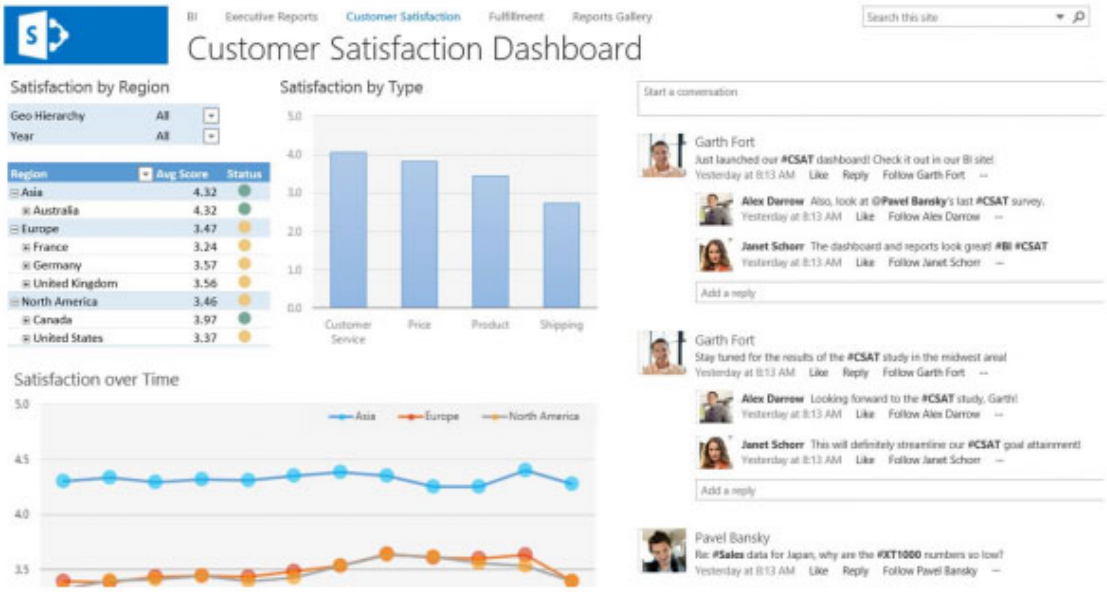

Figura 1.9: Cuadro de mando, creado por SharePoint, sobre la satisfacción de los clientes, discriminado por países o zonas, en diferentes aspectos como servicio al cliente y precio de los productos, entre otros.

7. **Área de Finanzas**. Análisis de gastos, rotación de cartera, razones financieras.
8. **Sector de Manufacturas**. Productividad en líneas, análisis de desperdicios, análisis de calidad, rotación de inventarios y partes críticas.
9. **Embarques**. Seguimiento de embarques, motivos por los cuales se pierden pedidos.

1.5. Desde los Datos hasta el Conocimiento

El objetivo de la inteligencia de negocios es generar información que pueda ser trabajada y analizada de forma intuitiva, más o menos en tiempo real, y con la posibilidad de integrar diferentes fuentes de datos para ofrecer una visión global que puede ser compartida y distribuida por todos los departamentos de la empresa. De manera que la idea no es que todo el mundo tenga acceso a toda la información, si no de que todo el mundo tenga acceso y genere la información que necesita para que su trabajo sea los más eficiente posible.

Para ello, la inteligencia de negocios debe pasar por las siguientes fases:

1. **Tratamiento de los datos en crudo**: Bases de datos, gestores de Bases de datos, etc.
2. **Obtener información a partir de los datos**: Herramientas de bases de datos, consultas y, principalmente, la **minería de datos**.
3. **Obtener conocimiento**: Visualización de la información y lectura adeacuada de la misma.

1.5.1. Tratamiento de los Datos.

Cuando los archivos están adecuadamente ordenados y mantenidos, los usuarios pueden acceder y recuperar fácilmente la información que requieren.

A pesar de existir, como veremos más adelante, sofisticadas infraestructuras y aplicaciones informáticas, muchas empresas no logran alcanzar esa información de valor cuando la necesitan. Una de las causas a las que es debido es que carecen de una solución de inteligencia de negocios que les permita acceder a los datos en cualquier momento y en cualquier lugar y que, a la vez, les permita manipularlos (filtrarlos, analizarlos, publicarlos, etc). Una aplicación de inteligencia de negocios debe poder acceder a datos de múltiples fuentes y tranformarlos en información valiosa de manera rápida y actualizada.

Un buen sistema de almacenamiento debe evitar:

- **Redundancias e inconsistencia de los datos**. Presencia de datos duplicados en varios archivos. Ocurre cuando distintos grupos en un organización recolectan por separado los mismo datos, que pueden generar confusión. Esto desperdicia recursos de almacenamiento y, también, conduce a la inconsistencia de los datos, donde un mismo atributo puede tener distintos valores.
- **Dependencia programa-datos**. Estrecha relación entre los datos almacenados en los archivos y los programas específicos que se requieren para actualizar y mantener tales archivos. Todo programa tradicional de cómputo tiene que describir la ubicación y naturaleza de los datos con que trabaja. En un entorno tradicional de archivos, cualquier cambio en los datos requiere un cambio en todos los programas que tienen acceso a los datos.
- **Seguridad Defectuosa**. Poco control de los datos, en el acceso a ellos y en la diseminación de la información. La gerencia tal vez no tenga forma de saber quién está accediendo a los datos de la organización, o incluso modificándolos.
- **Falta de compartición y disponibilidad de los datos**. Puesto que las piezas de información que están en archivos y partes difererntes de la organización no se pueden relacionar entre sí es prácticamente imposible que la información se comparta o se tenga acceso a ella de manera oportuna. La información no puede fluir libremente a través de las distintas áreas funcionales o partes de la organización.

Un sistema de administración de bases de datos (DBMS) resuelve estos problemas mediante software que permite su centralización y administración, de modo que las empresas tengan una sola fuente consistente para todas sus necesidades de datos. El uso de un DBMS minimiza la cantidad de archivos redundantes e inconsistentes.

Los principales gestores utilizados son los siguientes:

La mayoría de los **DBMS** tienen un lenguaje especializado conocido como lenguaje de manipulación de datos, siendo el más extendido el *lenguaje de consulta estructural* o *SQL (Structured Query Language)*.Una de sus características es el manejo del álgebra y el cálculo relacional que permite lanzar consultas con el fin de recuperar información de interés de una base de datos, así como también hacer cambios sobre la misma.

1.5.2. Sistemas de Almacenamiento de datos: Data Warehouse y Data Marts.

El concepto de **Data Warehouse** hace alusión al proceso mediante el cual una organización o empresa particular almacena todos aquellos datos e información necesarios para el propio desempeño de la misma. Se organiza con vistas a facilitar el análisis y la realización de reportes y consultas en las situaciones en que sea necesario hacerlo. Contar con un fácil acceso a los datos de importancia tendrá una relación directa con la efectividad de las diversas tareas de la empresa.

Si bien, tener un **Data Warehouse** no significa hacer inteligencia de negocios, y carecer de él tampoco significa no hacerla, este es un concepto base que facilita y fundamenta las demás técnicas y herramientas de la inteligencia de negocios.

Para poder alcanzar la correcta aplicación de la inteligencia de negocios es necesario utilizar técnicas de **Data Warehousing**. De manera, que su creación representa el primer paso, desde el punto de vista técnico, para implantar una solución completa y fiable de inteligencia de negocios.

La diversidad de sistemas transaccionales en las organizaciones hace que tener una visión unificada de los datos resulte ser una realidad muy compleja de alcanzar. Cada sistema presenta los datos con el enfoque para el que fue construido lo que termina por añadir mayor complejidad a la toma de decisiones

El objetivo de un **Data Warehouse** está enfocado a proveer una visión histórica y unificada de los datos de la empresa, de manera que resulte comprensible para los usuarios, utilizando una semántica esta expresada en los términos de negocio que ellos conocen

La información recogida en un **Data Warehouse** se basa en cuatro conceptos fundamentales:

- *Orientada hacia la información relevante de la organización*: Orientada a hechos, no a procesos. Se diseña para consultar información relativa a las actividades básicas de la organización (ventas, compras, producción,..), no para soportar los procesos que se realizan en ella (gestión de pedidos, facturación, etc.)

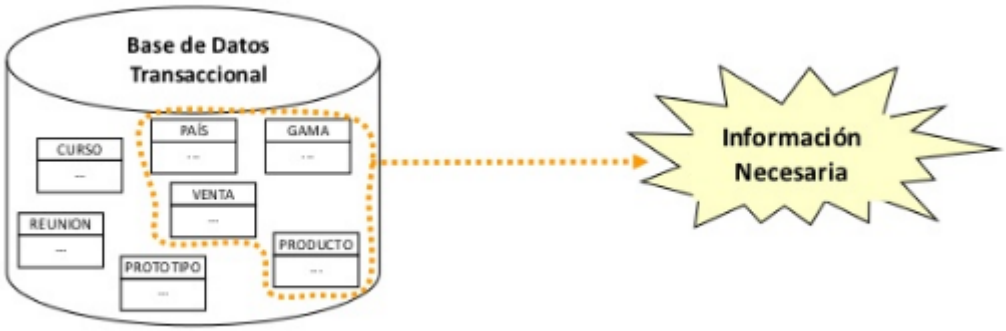

Figura 1.10: Información necesaria

- *Integrada*: Se integran datos recogidos de diferentes sistemas operacionales de la organización (y/ o fuentes externas)

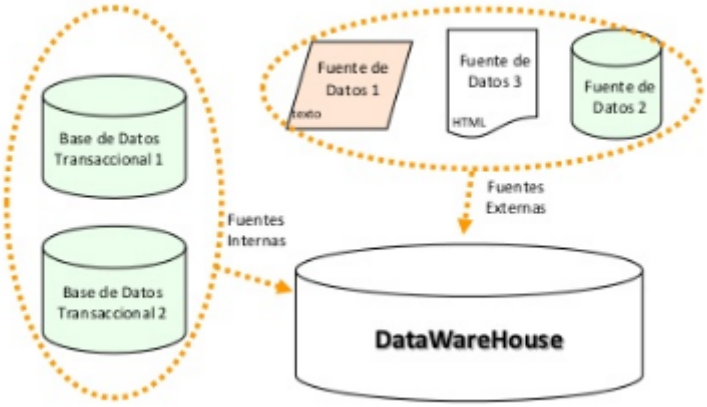

Figura 1.11: Datos integrados de diferentes fuentes

- *No volátil*: Los datos almacenadas no son actualiados, sólo son incrementados.

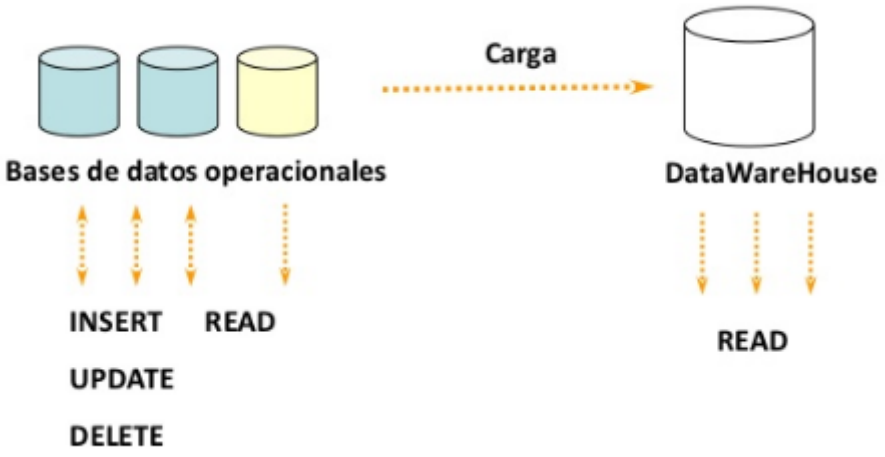

Figura 1.12: Información no volátil

- *Variable en el tiempo*: Almacena imágenes o estados de los datos (snapshots) en diferentes momentos del tiempo, reflejando así la *historia de los datos* a través del tiempo.

Es interesante ver, a través de la siguiente tabla, las diferencias en los sistemas transaccionales y los **Data warehouses** dentro de la estructura de inteligencia de Negocios:

Características	Sistemas Transaccionales (OLTP)	DataWarehouses (OLAP)
Datos	Actuales y Actualizables	Históricos y Estáticos
Almacenamiento	Bases de Datos Pequeñas y Medianas (MB y GB)	Bases de Datos Grandes (GB y TB)
Procesos	Repetitivos	No previsibles
Estructura	Detallada	Detallada con Niveles de Agregación
Usos	Soporte Operacional orientado a Procesos	Soporte de Análisis orientado a Información relevante
Unidad de Ejecución	Transaccional	Consultas
Cantidad de Datos	Miles	Millones
Modelo de acceso	Escritura Principalmente y Lectura Numero de transacciones Elevado	Lectura Principalmente Numero de transacciones Bajo o Medio
Tiempo de Respuesta	Segundos - Minutos	Segundos - Horas
Decisiones	Operativas Diarias	Estratégicas
Tipos de usuarios	Operativos	Administrativos
Número de usuarios	Miles o Mas	Cientos o Menos

Figura 1.13: Comparación DataWarehouses

Un concepto relacionado con el **Data Warehouse** es el de **Data Mart**. Un **Data Mart** es un repositorio de datos orientado a un área especifica del negocio, como recursos humanos, ventas o marketing.

Si lo comparamos con un **Data Warehouse**, es un almacen de datos más restringido por lo que requiere de menores costos y tiempo para construirse ya que maneja un menor volumen de datos.

1.5.3. Las Herramientas ETL

La **inteligencia de negocios** trata de homogeneizar y organizar la gran cantidad de datos almacenada en un *Data Warehouse*. Para ello se sirve de las llamadas **herramientas ETL**, *Extract, Transform and Load*, es decir "Extraer, Transformar y cargar". Su objetivo principal es desplazar todos esos datos procedentes de fuentes heterogéneas hacía un entorno de análisis homogéneo y tratable.

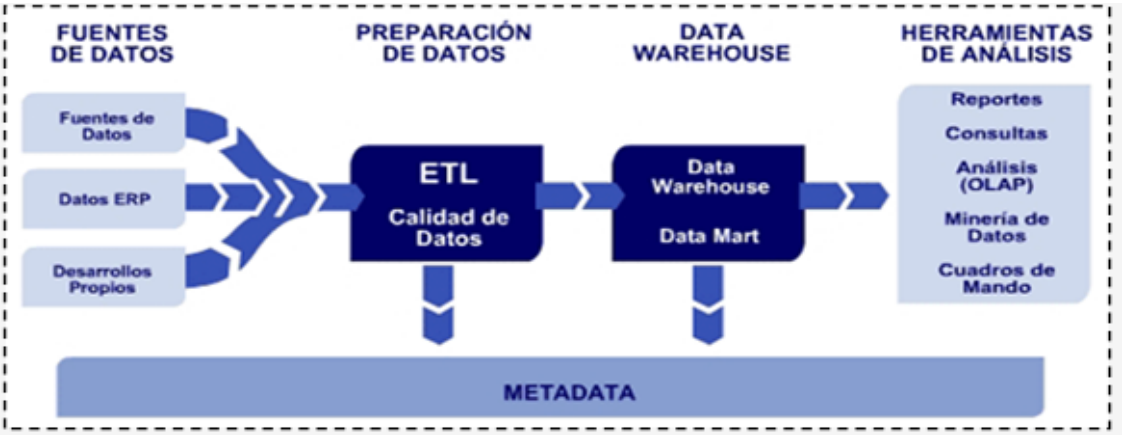

Figura 1.14: Esquema de Tecnologías e Información (TI) en la inteligencia de negocios

El proceso de funcionamiento se puede resumir en los siguientes pasos:

1. **Extracción** de los datos en crudo procedentes de fuentes de naturaleza distinta. Los datos están recogidos en fuentes heterogéneas y dispersas en el entorno de la empresa. La idea es utilizar aplicaciones de conectividad como ODBC, JDBC o procesos batch para extraerlos.

   Para la extracción se deben cumplir los siguientes aspectos:

   - No provocar cambios ni transformaciones sobre las fuentes de datos originales, de manera que su efecto sea el de menor impacto posible sobre ellos.
   - Criba de los datos que no aportan información relevante a los objetivos de la inteligencia de negocios.
   - Homogeneización de los datos a un formato común, facilitando la comparación e interpretación de la información.

2. **Transformación** de los datos extraidos en el paso anterior. Manipular, de manera automática, los datos según los fines y objetivos del negocio. Normalizamos los datos para analizarlos de manera apropiada.

   - Crea nuevos valores numéricos calculados o indicadores.
   - Agrupación o creación de totales.
   - Conversiones.

3. **Carga** de los "nuevos" datos tranformados en el conjunto estructurado de las bases de datos (*En el Data Warehouse*). Esta fase engloba:

   - Estructuración de los datos de manera jerarquizada para poder acceder a los niveles de información deseada. Es decir, organización de los datos de manera granulada.
   - Incorporación de restricciones de integridad, definición de rango permitidos para los datos según su naturaleza. Lo que significa que los datos tengan un sentido lógico en base a su significado intrínseco.

- Obtener una mayor calidad de los datos.

1.5.4. Obteniendo la Información

La **inteligencia de negocios** no es una tecnología, si no un conjunto de sistemas de información que trabajan de forma coordinada.

Entre las herramientas de bases de datos encontradas en el apartado de la inteligencia de negocios se encuentran:

1) Herramientas de procesamiento analítico, **Herramientas OLAP**.
2) Sistemas de **minería de datos (Data Mining)**.

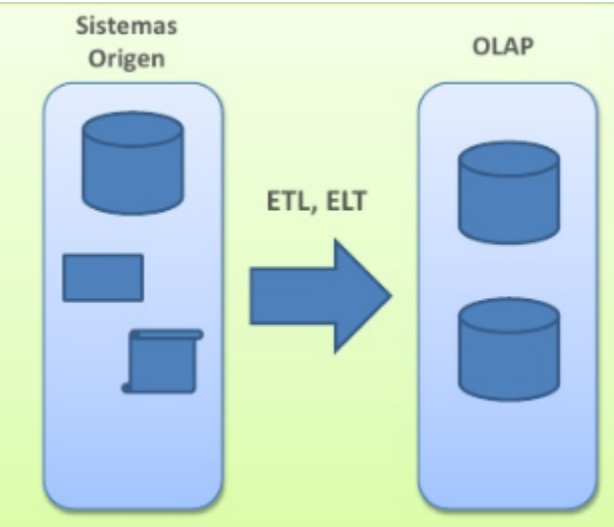

Figura 1.15: Esquema del procesamiento, transformación y preparación de los datos.

Las funcionalidades y objetivos de estas herramientas están altamente relacionadas entre sí. El **Data Warehousing** nos proporciona almacenamiento eficiente, con mantenimiento y recuperación de datos históricos. **OLAP** es un servicio que proporciona respuestas rápidas a consultas ad hoc a diferencia del **data warehouse**. Por otro lado, los algoritmos empleados en la **minería de datos** se encargan de encontrar patrones en los datos y de reportar modelos al usuario final, es decir, se trata, más propiamente dicho, del análisis estadístico - matemático al que se someten los datos, como por ejemplo realización de predicciones.

Vamos a tener un pequeño acercamiento a cada una de ellas.

1.5.4.1. Herramientas OLAP

OLAP (*Online Analytical Processing*), forma parte de lo que se conoce como sistemas análiticos que permiten responder preguntas como: ¿por qué paso?

Los sitemas **OLAP** se pueden considerar pertenecientes a los sistemas de información para ejecutivos (*EIS*), utilizados para proporcionar al nivel estratégico información útil para la toma de decisiones.

Los métodos OLAP surgieron para analizar los datos de ventas y marketing, así como para procesar datos administrativos y consolidar datos procedentes de diversas fuentes. El objetivo es efectuar diferentes análisis de aplicaciones que se caracterizan porque redefinen de manera continua y flexible el tipo de información que hay que extraer, analizar y sintetizar, por ejemplo de rentabilidad, mantenimiento de la calidad de los datos.

Son una estructura de base de datos multidimensional organizada de manera jerárquica. Los cubos **OLAP** pueden mostrar y sumar grandes cantidades de datos, a la vez que proporcionan acceso a los usuarios mediante búsqueda a los puntos de datos. De este modo, se puede resumir o reorganizar los datos según sea necesario, permitiendo procesar una variedad más amplia de preguntas pertinentes al área de interés del usuario. Son fáciles y rápidos de consultar, por lo que el acceso es más flexible para aquellos usuarios con un conocimiento técnico más limitado.

1.5.4.2. Data Mining

El **Data Mining**, también conocido como **minería de datos**, puede definirse, inicialmente, como un proceso de descubrimiento de nuevas y significativas relaciones, patrones y tendencias al examinar grandes cantidades de datos. Esto surge a partir de un proceso de *selección*, *exploración*, *modificación* y *modelización*. El proceso debe ser automático o semi-automático. El análisis de los datos se lleva a cabo a través de la utilización de tecnologías de reconocimiento de patrones, redes neuronales, lógica difusa, algoritmos genéticos y otras técnicas avanzadas.

Es una técnica clave para la inteligencia de negocios. Se podría decir que el término "inteligencia" de la **inteligencia de negocios** procede de la minería de datos.

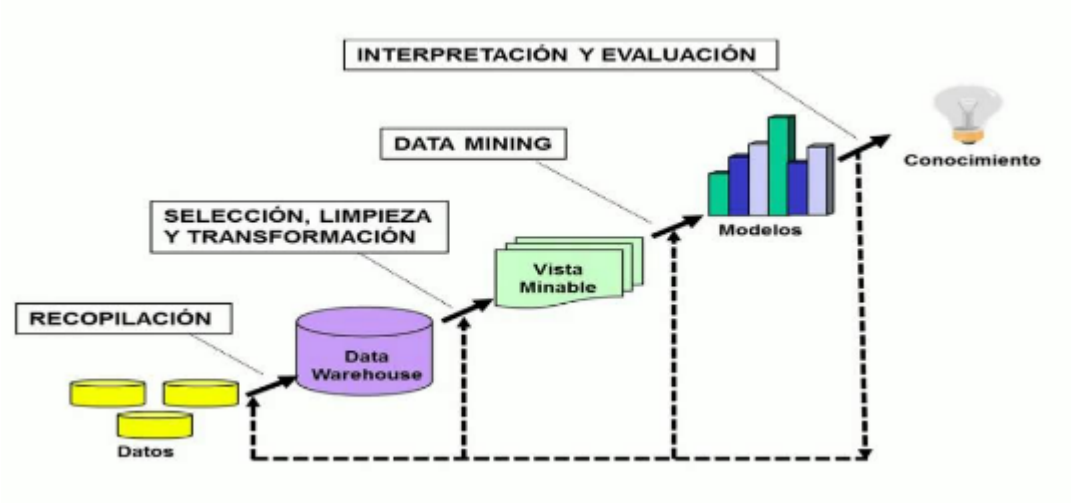

Figura 1.16: ¿Dónde se encuentra la minería de datos?

La **minería de datos** tiene incidencia en diferentes disciplinas como la *estadística*, la *inteligencia artificial*, los *aprendizajes de máquina*, el *reconocimiento de patrones*, etc. Ésta se basa en diferentes tipos de técnicas como *redes neuronales artificiales*, *árboles de decisión*, *algoritmos genéticos* y el método del *vecino más cercano*, entre otras. Algunos aspectos importantes que hay que comentar son:

- Es importante resaltar que, aunque es indudablemente importante descubrir y encontrar nueva información, es fundamental tener claro el significado de lo que estamos buscando y comprender los nuevos avances y resultados.
- Otro aspecto importante a considerar es la validez de los datos y, a raíz de ellos, el nivel de certidumbre de la información, ya que si la información que estamos analizando no tiene un grado considerable de formalidad podrán aparecer inconsistencias tanto en los datos como en las relaciones entre ellos.
- La información obtenida a partir de nuestra base de datos y mediante el uso de las diferentes técnicas que conforman la minería de datos, debe ser legible al usuario, este debe poder interpretar dicha información y encontrarle algún tipo de utilidad

1.5.4.2.1. ¿Qué problemas aborda la minería de datos?

Cualquier problema para el que existan datos históricos almacenados es un problema susceptible de ser tratado mediante técnicas de **Minería de Datos**, como pueden ser las siguientes:

- **Búsqueda de lo inesperado por descripción de la realidad multivariante**: Para describir un fenómeno cuantas más variables tengamos mejo; Obtenderemos descripciones más ricas, globales y coherentes, de manera que será más fácil detectar lo inesperado, aquello no previsto y que resulta valioso para entender mejor el comportamiento de algún grupo de individuos. Las muestras aleatorias son suficientes para describir la regularidad estadística global, pero no para detectar comportamientos particulares de subgrupos.
- **Búsqueda de asociaciones**: Resolver cuestiones a partir de un cierto suceso, del tipo: *"¿está asociado a otro suceso?"*, *"¿podemos inferir que determinados sucesos ocurren simultaneamente más de lo que seria esperable si fuesen independientes?"*, *"¿es posible sugerir un producto, sabiendo que otro ha sido adquirido?"*.
- **Definición de tipologías**: A efectos prácticos, los consumidores son entes individuales e infinitos, sin embargo, los distintos tipos de consumidores son muchos más pequeños y manejables. Detectar los distintos tipos de consumidores que podemos considerar, según su perfil de compra, y proyectarlos sobre toda la población es una tarea muy útil e imprescindible a la hora de plantear una política de marketing. Esta cuestión sobre las tipologías puede ser tanto de consumo, como de opiniones, valores, condiciones de vida, etc.
- **Detección de ciclos temporales**: Por ejemplo, todo consumidor sigue un ciclo de necesidades que repercuten distintos actos de compra a lo largo de su vida. Detectar estos diferentes ciclos o fases donde se sitúa cada consumidor facilitará adecuar la oferta de productos a las necesidades y crear fidelización de clientes.
- **Predicción**: Construcción de modelos a partir de datos hitóricos que nos permitan resolver cuestiones del tipo: ¿cuál es la probabilidad de baja de un cliente?, ¿cuál es el precio de una vivienda concreta?, ¿lloverá mañana? Si la variable respuesta es continua, como la rentabilidad de un cliente, se tratará de un problema de *regresión*, mientras que si la variable es categórica, como la compra o no de un producto, será un problema de *clasificación*.

1.5.4.2.2. KDD: Knowledge Discovery in Databases

El **Data Mining** se relaciona con un proceso conocido como **KDD**, *"Knowledge Discovery in Databases"*, que consta de las siguientes fases:

1) **Recopilación e integración** de la información a partir de unos datos iniciales (*fase de selección de datos*).
   - Fase fundamental para que, en las fases sucesivas, la extracción de conocimiento sea válido y útil a partir de la información original.
   - La información que se quiere investigar, generalmente, se encuentra en *bases de datos* y otras fuentes muy diversas. Se determinan las fuentes de información que pueden ser útiles y dónde conseguirlas.
   - El análisis posterior será mucho más sencillo y eficaz si la fuente está unificada, accesible y está desconectada del trabajo transacional, es decir, teniendo un almacén de datos con la información común y sin inconsistencias.
   - A parte de la información interna de la organización, los almacenes de datos pueden recoger información externa como demografías (censos), páginas amarillas, perfiles por zonas, uso de Internet, información de otras organizaciones o bases de datos externas compradas a otras compañías.
   - Se identifican y seleccionan las variables relevantes en los datos.

- La disponibilidad de grandes volúmenes de información en esta fase lleva a la necesidad de usar *técnicas de muestreo* para selección de datos.

2) **Preparación de los datos** (Exploración, limpieza o criba de datos y tranformación).

- Dado que los datos provienen de diferentes fuentes, es necesaria su *exploración* mediante técnicas de análisis exploratorio de datos, buscando, entre otras cosas, la *distribución de los datos*, su *simetría* y *normalidad*, y las *correlaciones* existentes en la información
- Para la exploración de datos se usan técnicas de análisis exploratorio de datos como los histogramas y los diagramas de caja y bigote, tallo y hojas que ayudan a detectar *datos anómalos* o atípicos(*outliers*). La presencia de datos atípicos y valores ausentes nos llevará a usar algoritmos robustos a datos atípicos y desaparecidos (*ej, árboles de decisión*), a filtrar la información y, o, a remplazar valores mediante *técnicas de imputación*.
- Si es necesario se llevan a cabo la transformación de los datos: tranformación de datos continuos en discretos mediante *técnicas de discretización*, técnicas avanzadas de transformación como las de *reducción* y *aumento de la dimensión*, *escalado simple* y *multidimensional*.
- Para la limpieza o criba de datos (*Data Cleaning*), deben eliminarse el mayor número posible de datos erróneos o inconsistentes (limpieza) e irrelevantes (criba). Se utilizan, casi exclusivamente, herramientas de consulta (*Query tools*) y herramientas estadísticas.

Estas primeras fases se engloban bajo el nombre de ***preparación de datos****.*

3) **Minería datos**.

- Se llevan a cabo a partir del desarrollo de modelos predictivos y descriptivos (*Model Development*) y mediante el análisis de datos (*Data Analysis*).
- Una vez recogidos los datos de interés se decide qué tipo de patrón se quiere descubrir. El tipo de conocimiento que se desea extraer va a marcar la técnica de **minería de datos** a utilizar. En esta fase en la que se decide que tarea se va a realizar (*clasificar*, *agrupar*, etc) y se elige la *técnica descriptiva* o *predictiva* que se va a utilizar.
- Es necesario seleccionar y validar los modelos anteriores, lo que consiste en el uso de *criterios de evaluación de hipótesis*. Se utilizan adicionalemnte *herramientas estadísticas* y de *visualización*. De manera que, si es necesario, se vuelve a las fases anteriores para una nueva interacción o modificación.

4) **Difusión y uso del conocimiento**.

- Los resultados pueden ser para muchos usuarios y necesitan difusión, con lo que se requiere que los usuarios sean expresados de una manera comprensible para ser distribuido en la organización. Es decir, se hace uso del nuevo conocimiento y se hace participe de él a todos los posibles usuarios.
- En esta fase se utilzan *herramientasde visualización*, *presentación* y *transformación de datos*.

Por lo tanto, la clasificación de las fases de la extracción del conocimiento podría resumirse de la siguiente manera:

1. **SELECCIÓN**

- Recopilar e integrar las fuentes de datos existentes.
- Indetificar y seleccionar las variables relevantes en los datos.
- Aplicar las técnicas de muestreo adecuadas.

2. **EXPLORACIÓN**

- Utilizar las técnicas de análisis exploratorio de datos.
- Deducir la distribución de los datos, simetría y normalidad.

- Analizar las correlaciones existentes en la información.

3. **LIMPIEZA**

- Detectar y tratar la presenecia de valores atípicos (outliers).
- Imputar la información faltante o valores perdidos.
- Eliminar datos erróneos e irrelevantes.

4. **TRANSFORMACIÓN**

- Utilizar técnicas de reducción y aumento de la dimensión.
- Aplicar técnicas de discretización y numerización.
- Realizar escalado simple y multidemensional.

5. **MINERÍA DE DATOS**

- Utilizar técnicas descriptivas: Clustering y Segmentación, escalamiento, reglas de asociación y dependencia, análisis exploratorio, reducción de la dimensión.
- Utilizar técnicas predictivas: Regresión y series temporales, análisis discriminante, métodos bayesianos, algoritmos genéticos, análisis de componentes principales, árboles de decisión, redes neuronales, máquinas de soporte vectorial.

6. **EVALUACIÓN E INTERPRETACIÓN DE RESULTADOS**

- Intervalos de confianza.
- Bootstrap.
- Análisis ROC.
- Evaluación de modelos.

7. **DIFUSIÓN Y USO DE MODELOS**

- Visualización.
- Simulación.

1.5.4.2.3. Técnicas del Data Mining

Las técnicas de la **minería de datos** tienen como fundamento a la *inteligencia artificial* y a la *estadística*. Estas técnicas no son más que algoritmos, más o menos sofisticados, que se aplican sobre un conjunto de datos para obtener unos resultados.

Pueden usarse varias técnicas a la vez para generar distintos modelos, aunque generalmente cada técnica obliga a un preprocesado diferente de los datos.

La clasificación inicial de las técnicas de minería de datos distingue las siguientes técnicas:

1) **Técnicas predictivas o de Aprendizaje supervisado**: Se basan en el entrenamiento de un modelo o método por medio de diferentes datos para poder predecir una determinada variable partiendo de estos mismos datos. Esta manera de trabajar se desarrolla en dos fases: entrenamiento (construcción de un modelo usando un subconjunto de datos como etiqueta, llamamos etiqueta al atributo del que vamos a predecir su valor) y prueba (prueba del modelo sobre el resto de los datos).

2) **Técnicas descriptivas**: En este caso no se asigna ningún papel determinado a las variables. También son llamados *métodos simétricos, no supervisados o indirectos*. Estos grupos con los que se trabaja no son conocidos con anterioridad, podemos encontrarnos con que las variables estén conectadas entre sí de acuerdo a vínculos desconocidos al principio. Esta opción es la elegida si la aplicación no es lo

suficientemente madura como para poder deducir una solución predictiva fiable. No se utilizan datos históricos en esta opción.

3) **Técnicas auxiliares**: Tanto las técnicas descriptivas como las predictivas buscan el descubrimiento del conocimiento de un conjunto de datos, pero también existen otras técnicas denominadas técnicas auxiliares que son herramientas de apoyo, las cuales están enfocadas más a la verificación.

1. TÉCNICAS PREDICTIVAS

- *Series temporales*: A partir de los datos hitóricos tenemos el comportamiento histórico de la serie, lo que permite modelizar las componentes básicas de la serie (tendencia, ciclo y estacionalidad) y así poder hacer predicciones para el futuro (cifra de ventas, previsión de consumo de un producto o servicio, etc). Interesa observar los cambios en esa variable a lo largo del tiempo y poder predecir valores futuros.

- *Regresión lineal*: Método estadístico que nos permite establecer una relación matemática entre un conjunto de variables predictoras y una variable dependiente. Se utiliza en aquellos casos en los que no se puede controlar los valores de las variables independientes Es la técnica más empleada para comparar datos. Es rápida y eficaz pero, en algunos casos, puede llegar a ser insuficiente en espacios multidimensionales donde pretendan relacionarse más de dos variables.

- *Análisis de la varianza y la covarianza*: Este análisis parte de los conceptos de la regresión lineal y permite eliminar la heterogeneidad causada en la variable de interés por la influencia de una o más variables cuantitativas.

- *Análisis discriminante*: Técnica estadística multivariante que ayuda a identificar las características que diferencian a dos o más grupos y a crear una función capaz de distinguir, con la mayor precisión posible, a los miembros de uno u otro grupo. Es capaz de decirnos que variables permiten diferenciar a los grupos y cuántas de estas variables son necesarias para alcanzar la mejor clasificación posible. Es una técnica muy parecida a la que se va a describir más adelante (Clustering) con la diferencia de que en ésta conocemos el número de datos y los datos que hay en cada grupo.

- *Métodos bayesianos*: Modelo estadístico en el que las observaciones se emplean para actualizar la probabilidad de que una hipótesis sea cierta o no. Es un método que necesita información anterior para determinar la distribución de probabilidad, que puede codificarse a partir del conocimiento apriori de un experto o puede ser inferido a partir de los datos. Se caracteriza por el uso constante del teorema de Bayes. Permite establecer relaciones causales y efectuar predicciones

- *Algoritmos genéticos*: Son métodos adaptativos que pueden usarse para resolver problemas de búsqueda y optimización que se inspiran en la evolución biológica. Se caracterizan por hacer evolucionar una población de datos sometiéndolas a acciones aleatoria semejantes y seleccionar cuáles son los individuos más adaptados y cuáles menos aptos.

- *Análisis de componentes principales (ACP)*: Técnica que se encarga de sintetizar la información y reduce la dimensión de las observaciones. Ante un banco de datos de muchas variables, se pretende reducirlas perdiendo la menor información posible. El resultado de esta reducción será una combinación lineal de las variables originales y serán independientes entre sí.

- *Árboles de decisión*: Son diagramas que representan de forma secuencial condiciones y acciones. Destacan por su sencillez y por poder utilizarse en distintas áreas, de manera que cualquier persona que no tenga grandes conocimientos puede entenderlo fácilmente.

  El objetivo es crear un modelo que predice el valor de una variable de destino en función de diversas variables de entrada. El mecanismo es elegir un atributo como raíz y desarrollar el árbol según las variables más significativas. Es decir, un *árbol de decisión*, es una forma gráfica y analítica de representar todos los sucesos que pueden surgir a partir de una decisión asumida en cierto momento, de manera que nos ayuda a tomar la decisión *"más acertada"*, desde un punto de vista probabilístico, ante un abanico de posibles decisiones. Los cuatro componentes requeridos en cualquier árbol de decisión son:

  - Alternativas de decisión en cada punto de decisión.
  - Eventos posibles tras cada alternativa de decisión.

  - Probabilidades de que ocurran cada evento posible.
  - Resultados de las interacciones entre las alternativas de decisión y los eventos.

- *Redes neuronales*: Esta herramienta emplea un conjunto de elementos de procesamiento de información altamente interconectados capaces de aprender con los datos de los que se abastece, tienen un cierto grado de "inteligencia". Es un procesamiento automático de aprendizaje inspirado en la forma en que funciona el sistema nervioso de los animales, por lo que reproduce algunas actividades del cerebro. Las características comunes entre las RRNN artificiales y las RRNN biológicas son el paralelismo masivo, la respuesta no lineal de las neuronas frente a la información recibida y el procesamiento de los datos recibidos a través de capas de neuronas.

  Estas capas de las que hablamos son tres: capa de entrada (recibe la información del exterior), capas ocultas (procesan la información internamente, no tienen ninguna conexión con el exterior) y capa de salida (obtiene la respuesta de la red dada por las capas ocultas y la transfiere al exterior).

- *Máquinas de soporte vectorial (SVM)*: Se usa para problemas supervisados de clasificación. Esta técnica está relacionada con la clasificación y la regresión. Se basa en un conjunto de algoritmos los cuales construyen hiperplanos en un espacio de dimensionalidad muy alta. Esto permite una separación de clases y una clasificación correcta. Potencialmente es capaz de generar muy buenos modelos predictivos.

Estas tres últimas técnicas, árboles de decisión, redes neuronales y máquinas de soporte vectorial, son de clasificación que pueden extraer perfiles de comportamientos, cuyo objetivo es construir un modelo que permita clasificar cualquier nuevo dato.

2. TÉCNICAS DESCRIPTIVAS

- *Clustering y Segmentación*: También denominado *análisis de conglomerados*, tiene un funcionamiento que se basa en clasificar una muestra de entidades en un número pequeño de grupos de forma que los elementos que formen un mismo grupo sean muy parecidos entre si (cercanos entre si) y muy distintos del resto de grupos. A diferencia del *Análisis Discriminante*, explicado anteriormente, en el Clustering no se conoce el número y la composición de los grupos. Las distancias más comúnmente utilizadas es la distancia euclidea, de Manhattan o de Chebyshev. También se utiliza como paso previo a otras técnicas en la minería de datos.

- *Escalamiento*: Representa en un espacio geométrico de pocas dimensiones las proximidades existentes entre los datos. Puede utilizarse como alternativa o como complemento a otras técnicas.

- *Reglas de asociación*: Se busca encontrar ítems que aparezcan juntos en transacciones de un determinado conjunto de datos. Para encontrar reglas de asociación hay que considerar todas las posibles combinaciones para que haya una consecuencia. Así, se establecen las reglas que indican dependencias entre los ítems de dicho conjunto de datos. Es decir, se pretende detectar cuándo la ocurrencia de un artículo está asociada a la ocurrencia de otros artículos en la misma transacción.

- *Reglas de dependencia*: Consiste en buscar un modelo que encuentre dependencias significativas entre el conjunto de datos. Estas dependencias pueden ser usadas para predecir valores futuros.

- *Análisis exploratorio*: La finalidad es conseguir un entendimiento de los datos y de sus relaciones. Permite organizar y preparar los datos, identificar casos atípicos y evaluar datos ausentes.

- *Reducción de la dimensión*: Tiene como objetivo reducir el número de variables aleatorias. Es utilizada cuando se tienen muchas dimensiones (atributos o variables) con respecto al número de instancias (casos), ya que pueden existir muchos grados de libertad.

3. TÉCNICAS AUXILIARES

- *Proceso Analítico de Transacciones (OLAP)*: Ya mencionado en apartados previos, es un proceso distinto a la minería de datos pero que se complementan, ya que no busca la creación de patrones a partir de los datos, si no que verifica estos patrones.
- *SQL y herrramientas de consulta*: Permiten aplicar el modelo a nuevos datos, obtener un resumen estadístico de los datos, realizar consultas de contenido, de predicción, de detalles y de definición de datos.
- *Reporting*: : Herramienta que permite crear, implementar y administrar informes donde se reflejan el análisis de los resultados de la minería de datos.

1.5.4.2.4. Proceso general del Data Mining

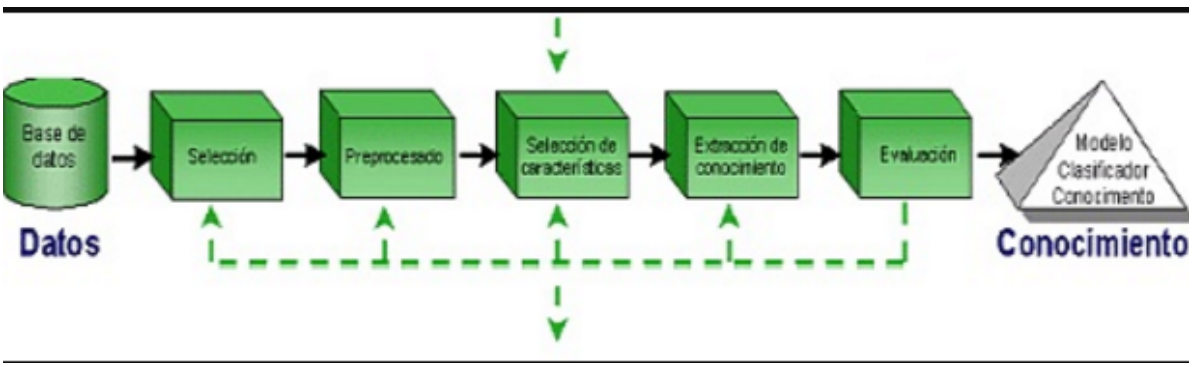

Figura 1.17: Esquema del proceso seguido en minería de datos.

- **Selección del conjunto de datos**. Tanto a lo referido a las variables objetivo (aquellas sobre las que se quiere predecir, calcular o inferir), como a las independientes (las utilizadas para el cálculo o proceso), como al muestreo de los registros disponibles.
- **Análisis de las propiedades de los datos**. Detectar la presencia de valores atípico, ausencia de datos y valores nulos, etc.
- **Transformación del conjunto de datos de entrada** (Preprocesamiento de los datos). Se realiza de diversas formas según el análisis previo, con el objetivo de prepararlo para aplicar la técnica de minería de datos que mejor se adapte a los datos y al problema
- **Seleccionar y aplicar la técnica de minería de datos**.Ya sea para construir un modelo de predicción, de clasificación o de segmentación. Por ejemplo, para la contrucción de un *modelo predicción* donde queremos estimar cuál será el comportamiento esperado del cliente, utilizamos los datos de uso de los servicios. En primer lugar, habría que *entrenar* el sistema usando datos de nuestros sistemas de información que contengan el comportamiento observado de los clientes.

  Después, por ejemplo, una vez al mes se realizaría un proceso de estimación del comportamiento previsto del cliente usando patrones de uso del servicio actualizados a este momento. Es decir, que con los datos actualizados del perfil de un usuario se puede predecir el comportamiento esperado del cliente, proceso al que se le denomina *scoring*.

Figura 1.18: Proceso de entrenamineto en Data Mining.

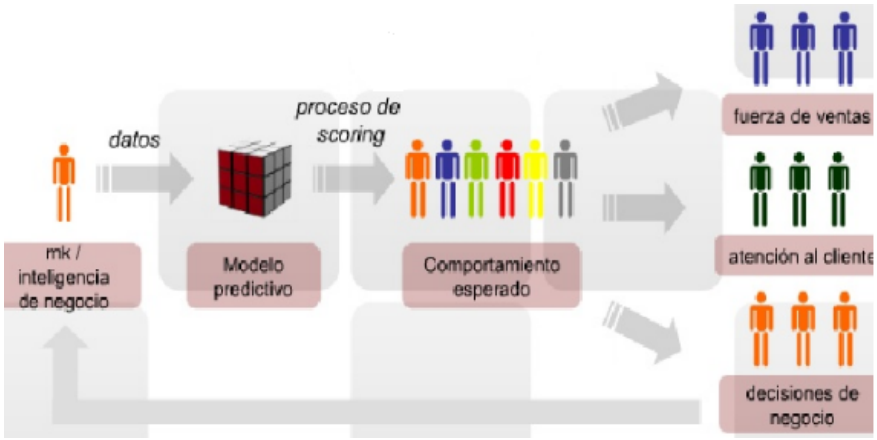

Figura 1.19: Proceso de scoring en Data Mining.

- **Extracción de conocimiento**. Mediante una técnica de minería de datos, se obtiene un modelo de conocimiento que representa patrones de comportamiento observados en los valores de las variables del problema o relaciones de asociación entre dichas variables. También pueden usarse varias técnicas a la vez para generar distintos modelos, aunque generalmente cada técnica obliga a un preprocesado diferente de los datos.
- **Interpretación y evaluación de datos**. Una vez obtenido el modelo se debe proceder a su validación comprobando que las conclusiones que arroja son válidas y suficientemente satisfactorias. En el caso de haber obtenido varios modelos mediante el uso de distintas técnicas han de comparase en busca de aquel que se ajuste mejor al problema. Si ninguno de los modelos alcanza los resultados esperados, debe alterarse alguno de los pasos anteriores para generar nuevos modelos.

1.5.5. Y llegó el Conocimiento.

El conocimiento es una mezcla de experiencia, valores e información, que sirve como marco para la incorporación de nuevas experiencias y es útil para la acción.

Para que la información se convierta en conocimiento es necesario realizar acciones como:

- Comparación entre diferentes elementos y aspectos de la empresa.
- Predicción de consecuencias que se pueden originar dada determinadas acciones.
- Búsqueda e identificación de conexiones, relaciones y patrones.
- Conversación con otros portadores de conocimiento, de manera que se retroalimente la información.

Para alcanzar este aspecto necesitamos de un conjunto de herramientas que nos permitan explorar la información. Este el área donde más avances se han producido en los últimos años. Podemos nombrar tres herramientas básicas:

1) **Query and Report**. Las herramientas de **Query** (consulta) y **Report** (informe) son fundamentales para el desarrollo y manejo de listados e informes basados en la información presente en los almacenes de datos y **Data Marts**.
   - *Consultas*: A través de la realización de múltiples consultas de las bases de datos podemos obtener un análisis más profundo de la información, ya que tenemos todas las bases de datos interconectadas.
   - *Informes*: Algunos informes dan una visión estática de los resultados y su análisis, de manera que no permiten interacción con ellos como, por ejemplo, conocer que resultados obtendríamos ante el planteamiento de diferentes situaciones o escenarios. Lo interesante de la inteligencia de negocios es que puede estar enlazada con los conjuntos de datos originales, lo que permite manipular los datos dentro del informe.
2) **Cuadro de Mandos Analítico**: Se suelen emplear en la inteligencia de negocios, ya que proveen a los gerentes de un acceso sencillo a la información relevante, interna y externa, de su compañía. Se generan a partir de datamarts, de informes resumen e indicadores clave para la gestión (*KPI*). Estos permiten a los directivos o usuarios analizar los resultados de la empresa de forma rápida y eficaz. En la práctica, es una herramienta de consulta, *query*, orientada a la obtención y presentación de indicadores para la dirección (frente a la obtención de informes y listados).
3) **Cuadro de Mandos Integral o Estratégico** (Balanced Scorecard): Con este tipo de cuadro de mandos, los diferentes niveles de dirección y gestión de la organización disponen de una visión de la estrategia de la empresa traducida en un conjunto de objetivos, iniciativas de actuación e indicadores de evolución.

Mediante ambos cuadros de mando, se presentan un conjunto de indicadores y representaciones gráficas que dotan a la empresa, y a sus usuarios, de una visión de la situación del negocio. A través de estas visualizaciones de datos obtendrán una radiografía de lo qué ocurre tanto de manera interna como externa a la compañía.

A partir de la estructura del conocimiento obtenemos una ruta de navegación que permitirá a la empresa delinear nuevas políticas. Sin embargo, como todo aspecto que conlleve una interpretación visual, se corre el peligro de derivar en un análisis incorrecto, lo que podría incluso originar el fracaso de la empresa. Por esta razón, es fundamental que los gráficos e indicadores que representen la información de la empresa sean claros y de una lectura sencilla y completa de la realidad de la empresa.

Esta claridad necesaria sobre las representaciones gráficas e indicadores, la podemos ver en la figura 1.20, donde podemos ver un **cuadro de mandos** realizado con **Qlik Sense**.

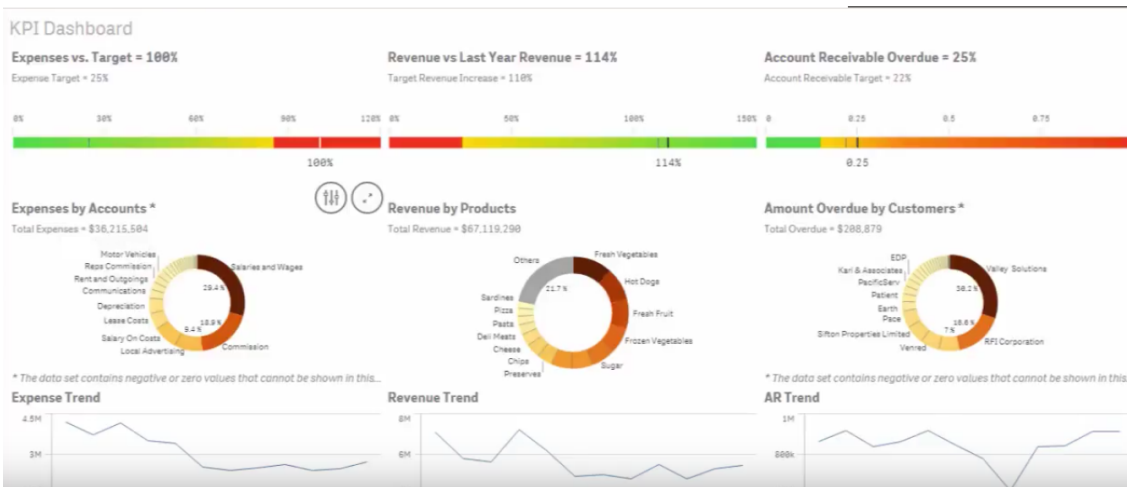

Figura 1.20: Dashboard con indicadores.

Como puede verse en la figura 1.20, tenemos unos indicadores que representan el porcentaje de diferentes aspectos de la empresa. Presenta el porcentaje real frente al objetivo marcado para ese aspecto por la propia empresa.

Figura 1.21: Indicadores mediante porcentaje y color.

En la figura 1.21, primero vemos que el objetivo, referido a los gastos, es de un 25 % y nos encontramos en un 100 %, por lo que el indicador se situa en la zona roja de la barra. De forma análoga para los otros dos.

Si nos fijamos en la figura 1.22, observamos un gráfico circular correspondiente a los gastos y podemos detectar que partida es la que representa mayor gasto. Vemos que es la referente a salarios y nóminas.

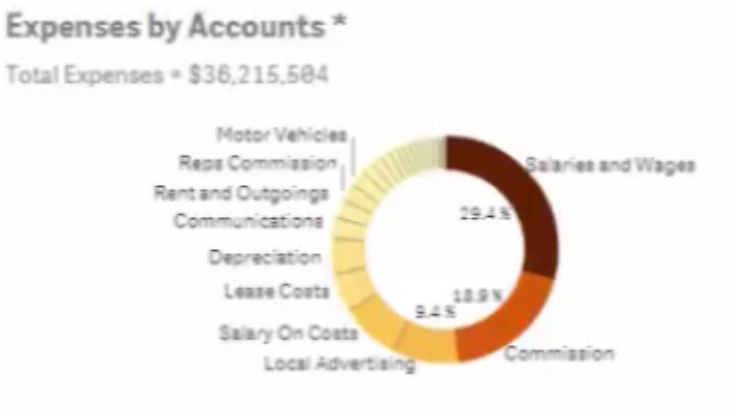

Figura 1.22: Porcentajes de gastos por secciones.

Finalmente tenemos gráficos de tendencias, donde se nos muestra la evolución temporal de un indicador.

Tras la lectura de este **cuadro de mandos** podemos ver que no siempre se alcanza esa sencillez a la hora de realizar la interpretación de los indicadores y gráficos que aparecen. En este ejemplo hemos podido ver que, sin una breve explicación de lo que se realiza y se muestra, no habríamos sido capaces de tener un razonamiento acertado sobre lo que se pretende mostrar.

Por lo tanto, todos los gráficos pensados para ser usados en un ámbito de la inteligencia de negocios deben ser diseñados con claridad y simplicidad, de manera que el mensaje que se quiera hacer llegar al usuario sean expresesados de manera concisa y clara.

1.5.5.1. Representaciones gráficas

Las herramientas más utilizadas y útiles para analizar información son las representaciones gráficas.

Fácilmente, a través de los gráficos, podemos transmitir de manera instantánea y sistemática las relaciones que pueden existir entre los datos. Sin embargo, en ocasiones, se utilizan indistintamente los distintos tipos de gráficos existentes, sin tener en cuenta la naturaleza del problema que queramos representar.

De esta forma, el objetivo principal y fundamental del uso de los gráficos debe ser *"transmitir la máxima información de forma más directa y sencilla, para que el usuario descubra relaciones entre los datos, obtenga respuestas y tome decisiones"*.

Los gráficos más comunes que se utilizan en inteligencia de negocios son los siguientes:

1) **Gráfico de columnas / barras**. Tanto el gráfico de columnas, como el de barras, son los más usuales a la hora de realizar informes. La longitud de cada columnna, orientadas verticalmente como podemos ver en la figura 1.23, es proporcional al valor que representa. Se usan para comparar valores y, a veces, las columnas se representan con una línea en vez de un rectángulo. Los gráficos de barras presentan una horientación horizontal.

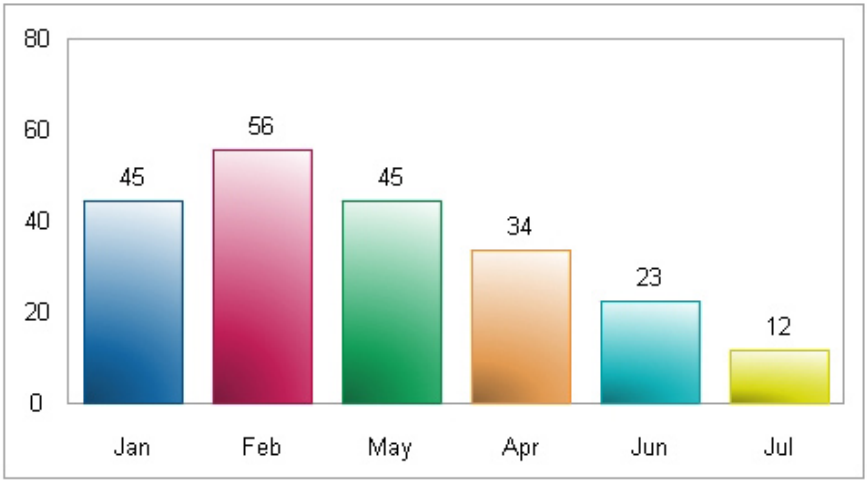

Figura 1.23: Gráfico de columnas, comparando cantidades en los diferentes meses.

2) **Gráficos de "líneas y puntos"**. Creado para conectar una serie de puntos que representan medidas individuales a través de segmentos. Se suele utilizar para representar tendencias, mostrando la evaluación a lo largo del tiempo. Los gráficos de puntos se centrarán en mostrar la posición "actual" que tiene cada dato.

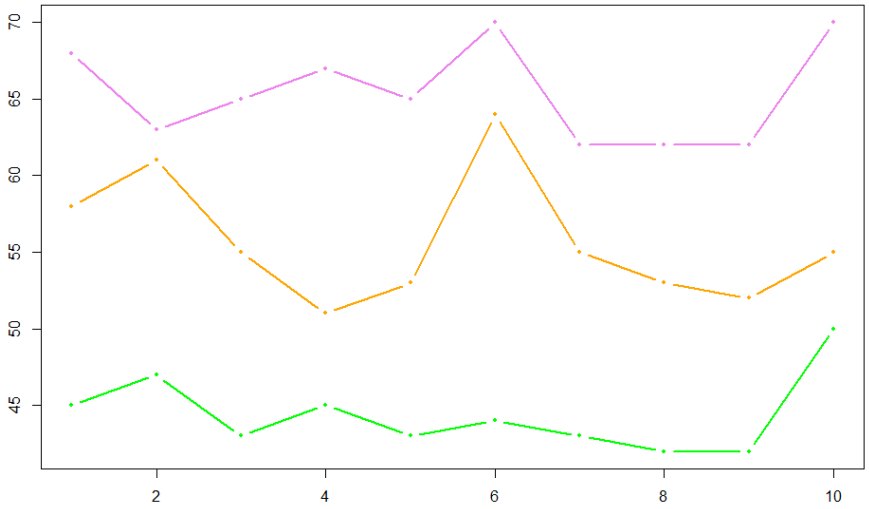

Figura 1.24: Gráfico de líneas, para obtener una rápida visualización sobre los datos.

3) **Gráficos de sectores**. Gráficos circulares divididos en secciones, que muestran la magnitud relativa o frecuencias de una variable. En los gráficos de sectores, el área de cada sector es proporcional a la cantidad que representa. Como podemos ver en la figura 1.25, si añadimos un *hueco* al gráfico se le denomina *donut chart*.

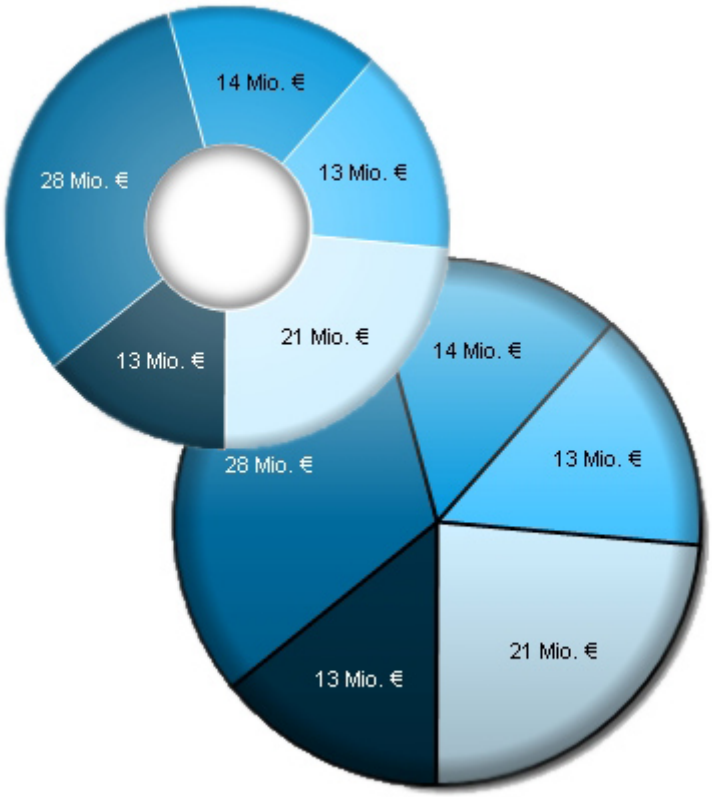

Figura 1.25: Gráfico de sectores y donut.

4) **Gráficos de Área**. Gráfico basado en la representación de líneas que muestra gráficamente datos

cuantitativos. El área entre el eje y la línea es normalmente rellenada de color, como vemos en la figura 1.26, con texturas o sombreados. Normalmente, están especialmente indicadas para comparar dos o más cantidades, pero también se usan para representar totales acumulados, utilizando números o porcentajes.

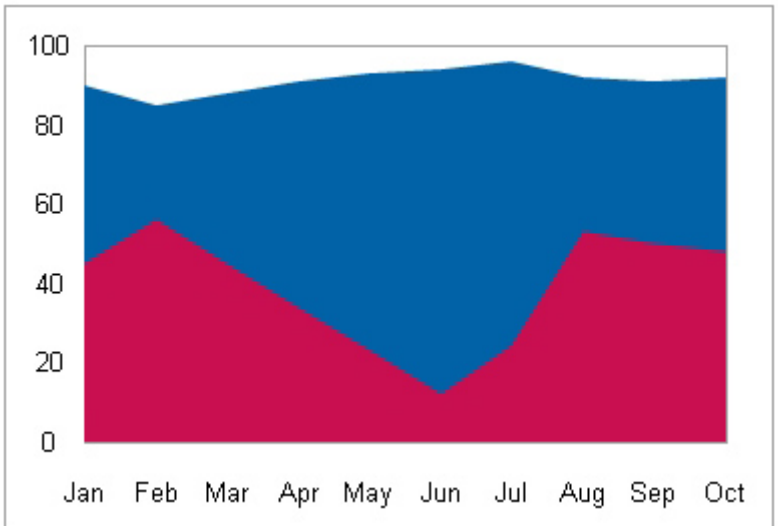

Figura 1.26: Gráfico de áreas que representa variables cuantitativas.

5) **Gráficos de Burbujas**. Representan la relación entre tres medidas, utilizando el eje X e Y y el tamaño de las burbujas. Suelen ser utilizadas para representar datos financieros. Cuando los valores de la tercera dimensión son muy diferentes (como podemos ver en la figura 1.27), las burbujas son una forma muy adecuada de representar estas relaciones.

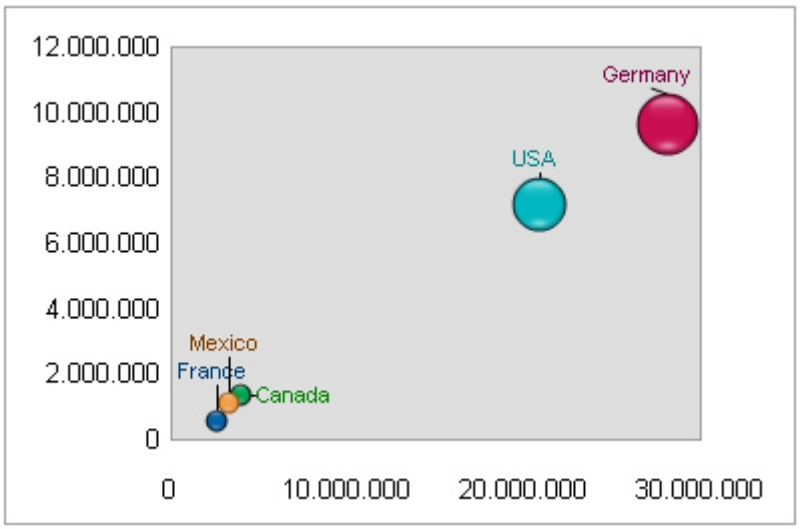

Figura 1.27: Gráfico de burbujas que representa 3 dimensiones de variables.

6) **Gráficos de Radar o "spider"**. Se utilizan para representar datos multivariantes (figura 1.28) en un grafico bidimensional para tres ó más variables cuantitativas.

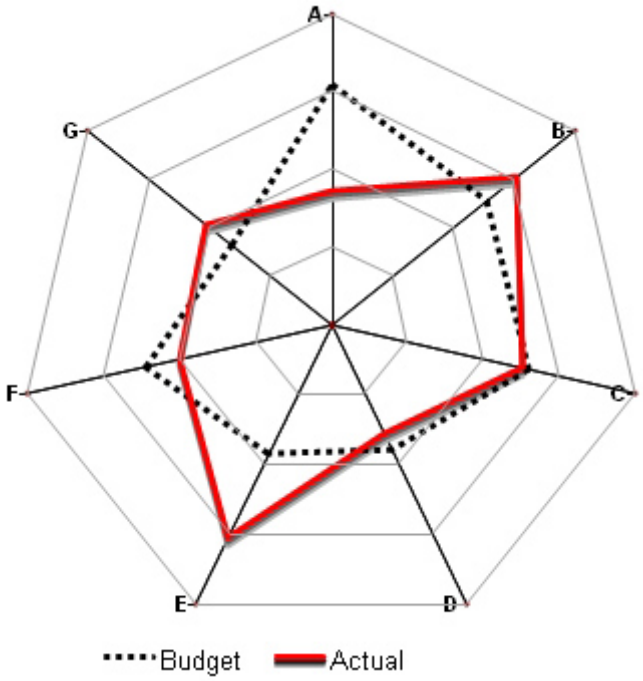

Figura 1.28: Gráfico de radar que representa facilmente datos multi-dimensionales.

7) **Mapas**. Representan variables, por ejemplo número de clientes o población, categorizada mediante algún atributo (como colores, como podemos ver en la figura 1.29, o formas) sobre el mapa, indicando la región geográfica.

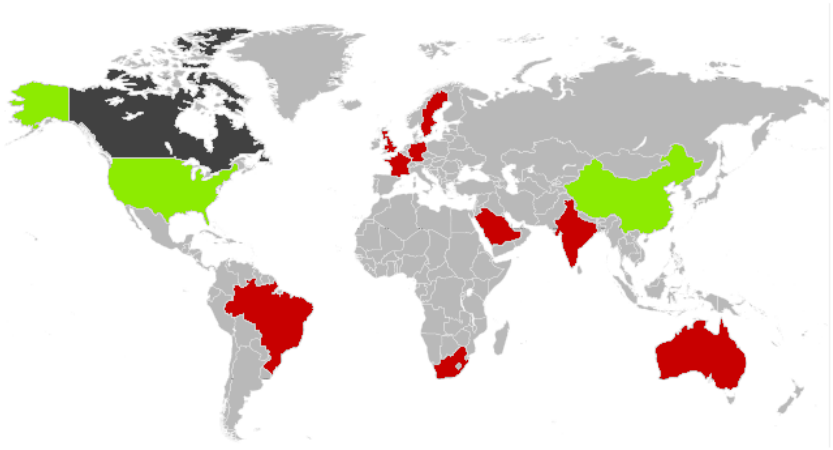

Figura 1.29: Gráfico de mapa que representa los datos en un contexto geográfico.

8) **Gauges o Velocímetros**: Muestran uno o más valores usando indicadores y una visualización de métricas adecuada, colocadas dentro de una especie de *velocímetro*, como podemos ver en la figura 1.30. Se suele utilizar en los **cuadros de mandos** para resaltar algunos indicadores de rendimiento.

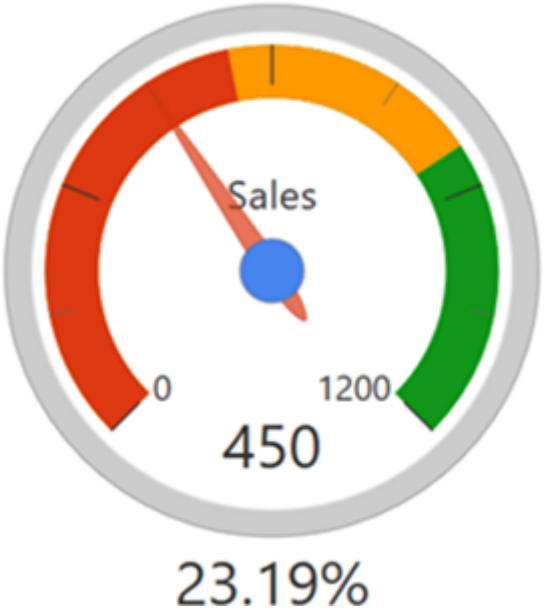

Figura 1.30: Gauge de ventas

Estos, son solo algunos de las representaciones gráficas más habituales que podemos encontrar en un **cuadro de mandos**. Sin embargo hay muchas más posibilidades, y más sofisticadas, para obtener el **cuadro de mandos** que mejor se adapte a nuestras necesidades coorporativas. Podremos encontrar más información sobre ellos y sobre cómo implantarlos en un cuadro de mandos en los capítulos siguientes.

1.6. Cuadros de Mandos

1.6.1. Aspectos fundamentales para crear y diseñar un buen Cuadro de Mandos.

Una vez que ya hemos realizado las etapas correspondientes al análisis de los datos y su oportuna conversión en información útil, llegamos a la fase de visualizar y presentar todo ese conocimiento que hemos podido extraer.

Es aquí donde entra en juego la creación y diseño de un buen **dashboard** o **cuadro de mandos**. La forma de hacer que nuestro trabajo brille, y la información cobre valor, es que su visualización esté bien hecha.

A continuación, vamos a presentar algunos aspectos fundamentales a tener en cuenta a la hora de pensar, crear y diseñar nuestro **cuadro de mandos**

- Es interesante explicar tanto el qué ha pasado como el porqué. Por ejemplo, mostrar un dato informativo sobre el número de visitas que recibe la página web pero acompañándolo del crecimiento respecto a los meses anteriores.
- Buscamos mostrar datos sencillos que hablen de manera elocuente sobre la situación de la empresa.
- El objetivo es mostrar toda la información de manera que pueda ser revisada de un vistazo, idealmente en una única pantalla. Lo interesante es mostrar entre 4 ó 5 KPI's (*Key Perfonmance Indicator*), o indicadores de rendimiento, muy claros que ayuden al negocio.
- La información que se muestra mediante los KPIs debe ser, únicamente, aquella que lleva a la toma de decisiones y a la "acción".

- El cuadro de mandos debe permitir la interacción y adaptación al usuario ya que, como ya hemos comentado anteriormente, cada usuario, incluso de un mismo departamento, puede requerir de diferente información. Se busca tener toda la información en un cuadro de mandos y que el usuario pueda interactuar.
- Queremos que haya el mínimo ruido o distracción posible. Un error común es intentar mostrar e introducir la mayor cantidad de datos posibles en el cuadro de mandos. Con esto, lo único que conseguimos es debilitar la efectividad del cuadro de mandos.
- Obtener una visión global pero que permita una incisión más detallada en caso de que el usuario lo requiera. Por ejemplo si nuestro cuadro de mandos presenta las devoluciones totales debe permitir detallar, por ejemplo, que productos son los que han sido devueltos.
- El uso de colores y formas permite *contar*, de manera muy intuitiva y visual, qué esta ocurriendo. Por ejemplo, datos que son perdidas señalarlas en rojo para que llamen la atención.

Table 1

Customer #	Customer Name	Salesman	Country	YTD Sales ($)
100007	A-1 Cycle City	Amy Schilling	United States	840677
100004	A/B Cycle Shoppe	Rick Sterling	United States	700136
100062	Accessory Zone	Sally Peterson	United States	686420
100005	Ace Hardware	Sal Stangarone	United States	548345
102053	Action Bikes	Scott Carlson	United Kingdom	123894
100031	Advanced Bicycles	Rick Sterling	United States	548340
102004	Aires Cycles	Patricia Renolds	Argentina	700136
100100	All Weather Cycles	Rick Sterling	United States	540339
102036	Alton Bicycles	Michael Coleman	Korea	229202
101051	Amardeep Cycles	Brian Kurtzman	India	456440

Table 2

Customer #	Customer Name	Salesman	Country	YTD Sales ($)
100007	A-1 Cycle City	Amy Schilling	United States	840677
100004	A/B Cycle Shoppe	Rick Sterling	United States	700136
100062	Accessory Zone	Sally Peterson	United States	686420
100005	Ace Hardware	Sal Stangarone	United States	548345
102053	Action Bikes	Scott Carlson	United Kingdom	123894
100031	Advanced Bicycles	Rick Sterling	United States	548340
102004	Aires Cycles	Patricia Renolds	Argentina	700136
100100	All Weather Cycles	Rick Sterling	United States	540339
102036	Alton Bicycles	Michael Coleman	Korea	229202
101051	Amardeep Cycles	Brian Kurtzman	India	456440

Figura 1.31: Tabla sin colores, y con colores. Desde tonalidades verdes (positivo), hasta un gradiente de amarillo, naranja y rojo (aviso, precaución, peligro).

- No podemos olvidar la estructura, o arquitectura, de fondo con la que hemos creado nuestro cuadro de mandos. Esta determina como de fácil, o no, será integrar el cuadro de mandos en otro programa, o si podremos o no mover nuestro cuadro de mandos a una localización diferente, como a un servidor en la nube. Es decir, la arquitectura va a determinar la capacidad de adaptación al cambio de tecnología.
- Debe presentar información en tiempo real. Si la información que se presenta en nuestro cuadro de mandos no puede ser usada para manejar o solucionar los problemas actuales, o mejorar la situación actual, no es un cuadro de mandos si no un informe. No solo se deben mostrar *históricos*.

Además, un **cuadro de mandos** sugiere varias perspectivas según el fin que se pretende:

1. **Desarrollo y Aprendizaje**. Indicadores que dotan a la organización de la habilidad para mejorar y aprender. Algunos indicadores son:
   - Capacidad y competencia de las personas.
   - Satisfacción de los empleados.
   - Productividad.
   - Necesidad de información.
2. **Procesos**. Analiza los procesos internos de la empresa de cara a la obtención de la satisfacción del cliente y logro de altos niveles de rendimiento financiero. Algunos indicadores son:
   - Tiempo de Servicio.
   - Índice de reclamaciones.
   - Calidad de productos.

3. **Clientes**. Se miden las relaciones con los clientes. Algunos indicadores son:
   - Índice de Satisfacción.
   - Índices de visitas Web, Facebook...
   - Índice Pérdida de clientes.
   - Tipo de Origen de clientes.
   - Nivel de conocimiento de Marca.
4. **Finaciera**. Están basados en la contabilidad de la empresa, y muestran datos históricos. Algunos indicadores son:
   - De liquidez.
   - De endeudamiento.
   - De rendimiento del capital invertido.

Visto todos estos aspectos, podemos concluir que para que un cuadro de mandos sea eficiente y transmita la información correctamente, debe de cumplir la regla de las *3 grandes W*. [6]

1. **WHAT** ¿Qué está pasando?: Mostrar qué ocurre a través de KPIs y métricas dentro de un contexto, comparando periodos y, o, tendencias.
2. **WHY** ¿Por qué está pasando?: Hay que informar de los motivos.
3. **WHAT NOW?** ¿Y ahora qué? ¿Cómo podemos arreglarlo o mejorarlo?: Mostrar soluciones y mejoras.

1.6.2. Ejemplos de Cuadros de Mando en Ámbitos Reales

1.6.2.1. Cuadros de Mandos en Educación: Navegador de Información sobre Escuelas Italianas.

Aquí podemos ver un ejemplo desarrollado con el fin de obtener una descripción del sistema educativo italiano a través de diferentes elementos que lo definen: Institutos, profesores, alumnos, costes, etc. Los siguientes cuadros de mandos han sido construidos con la aplicación **Qlik View**.

A través de esta aplicación podremos descubrir información relevante, comparar y analizar los datos, etc.

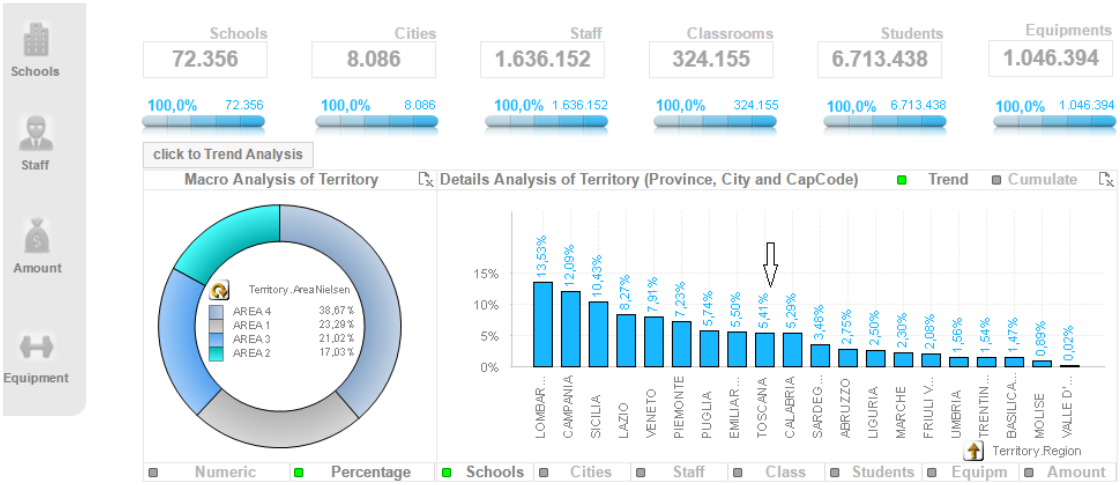

Figura 1.32: .

[6] http://ocalab.es/preocupate-por-disenar-un-buen-dashboard-y-obtendras-un-gran-reporting/.

En la figura 1.32, podemos ver información relativa al número de estudiantes por cada región. Podemos hacer "zoom" en la región de *La Toscana* para ver información sobre ella.

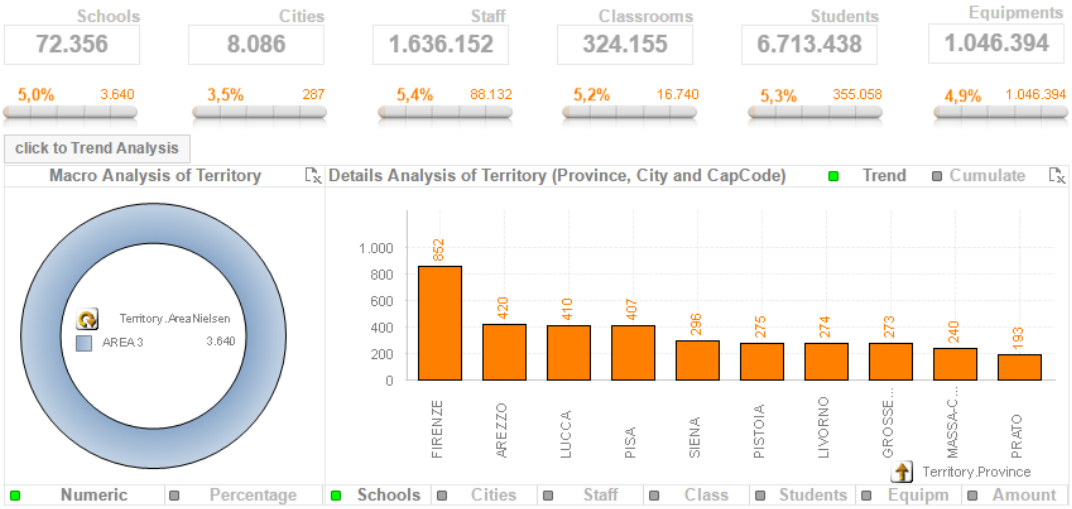

Figura 1.33: .

En la figura 1.33, vemos la información detallada de las provincias que se encuentran en La Toscana.

1.6.2.2. Cuadros de Mandos en Sanidad: Visión sobre la sala de emergencia de un hospital

Evidentemente, los profesionales de la salud, y las propias organizaciones responsables, buscan brindar un servicio médico de alta calidad a los usuarios. Por ello, uno de los aspectos más importantes sobre el que deben prestar una gran atención y esfuerzo es a la sala de emergencia.

A continuación vamos a ver un **cuadro de mandos** [7] que proporciona alguna de los índices y métricas, que se actualizan en tiempo real, del estado de la sala de emergencia sobre los cuales los trabajadores necesitan conocer y estar informados.

[7] https://www.klipfolio.com/resources/dashboard-examples/healthcare/er-status

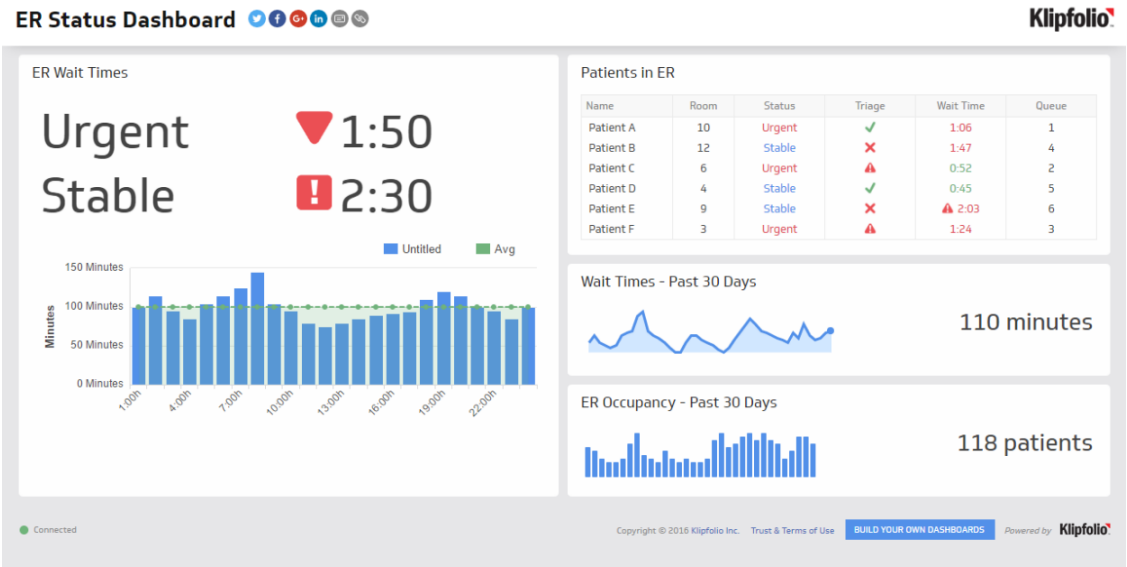

Figura 1.34: Cuadro de mando que proporciona a los trabajadores una serie de métricas e indicadores en tiempo real sobre la sala de emergencias.

Los aspectos que podemos ver en la figura 1.34 son los siguientes:

- Monitorización del tiempo de espera de los pacientes en la sala: Por horas en el día actual.
- Monitorización del estado de los pacientes: Estables y "en peligro".
- Evolucion en los últimos 30 días: Número de pacientes y tiempos de espera.

1.6.2.3. Google Analytics

Google Analytics [8] es un servicio gratuito de analíticas para cualquier sitio web con decenas de herramientas de estadísticas y de análisis, además de informes predeterminados y personalizables según lo requieran los usuarios de la información.

Ofrece una amplia gama de mediciones y análisis, algunas de ellas son:

- Cuantía y duración de las visitas.
- Datos sociodemográficos de los usuarios (lenguaje, ubicación, proveedor de Internet).
- Registro del comportamiento de los usuario dentro del sitio web (fuentes de tráfico, páginas visitadas, secciones preferidas, desplazamientos entre ellas, palabras clave usadas).
- Análisis del tiempo de carga.
- Análisis de redes sociales, registro del contenido más popular, detalles técnicos de los dispositivos de los visitantes (navegador, OS, referencia del móvil utilizado para acceder).
- Registro de conversiones, aspecto muy útil para sitios de ecommerce (ventas, descargas, reproducciones de videos, etc.).
- Múltiples gráficos estadísticos y muchas funciones más.

[8] https://www.google.es/intl/es/analytics/

Ejemplo Cuadro de mandos Geográfico

Especialmente, si se está interesado en *comercialización local*, posiblemente queramos saber de dónde son los usuarios que visitan nuestra página web.

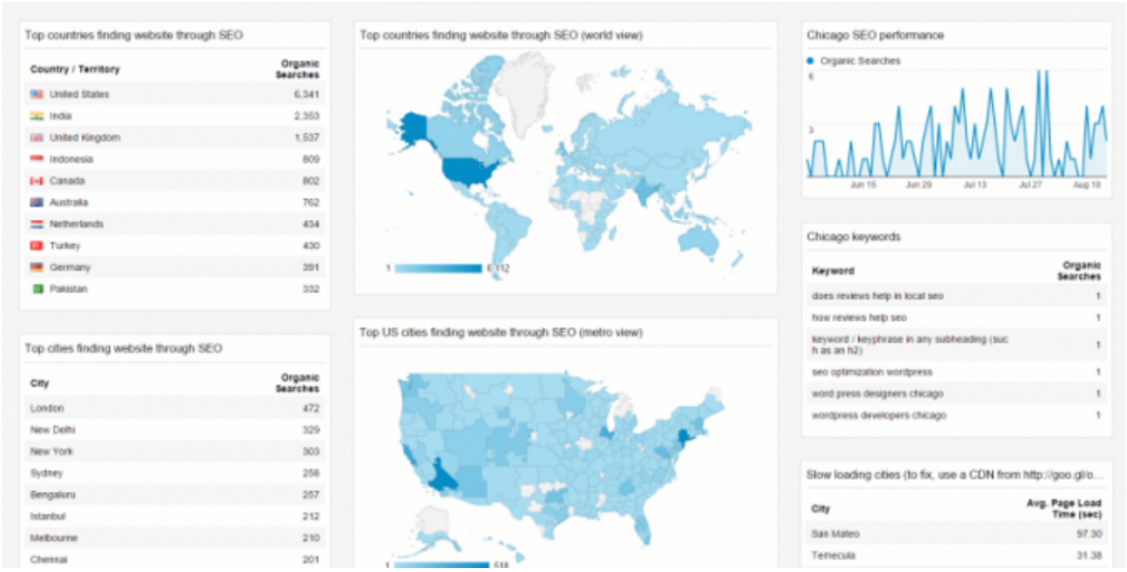

Figura 1.35: Cuadro de mando Geográfico Google Analytics.

A través del uso de filtrado en el mapa haremos referencia a una parte específica de los datos sobre los visitantes.

A través de este cuadro de mandos, que podemos ver en la figura 1.35, podemos obtener los visitantes por ciudad, estado, o continente y las ciudades que presentan un tiempo de carga de la web más lenta, por ejemplo, quizás, querramos hacer nuestra web más rápida para estos visitantes.

Capítulo 2

Principales Aplicaciones Informáticas para la Inteligencia de Negocios

2.1. Introducción

Existen múltiples aplicaciones informáticas que permiten incorporar la **inteligencia de negocios** al manejo de la empresa o negocio. En este capítulo vamos a ver algunos de los principales programas comerciales que podemos encontrar actualmente.

Hay una gran variedad de aplicaciones de inteligencia de negocios de diversos tipos de fabricantes, inclusive los llamados Open Source. La gran oferta de productos ha originado una estrategia comercial muy atractiva para los consumidores, incluso más de un prestigioso fabricante ofrece versiones gratuitas de su producto estrella. La selección de una herramienta de BI implica tanto factores racionales como subjetivos entre los que se podrían contemplar los siguientes:

- Total inversión económica.
- Disposición de personal cualificado.
- Facilidad de uso.
- Casos de éxito locales.
- Popularidad, por consiguiente, fuentes de referencia para compartir experiencias.
- Soporte que brindan los fabricante o partners.
- Servicios y productos adicionales (formación y herramientas gratuitas extras).
- Soportes de entrada y salida permitidos (Web, Móvil. . .).
- Posicionamiento del fabricante en informes de consultoras especializadas (IDG o Gartner).

2.2. Características principales que toda aplicación de Inteligencia de Negocios debe tener

Las plataformas de inteligencia de negocios se caracterizan por ser herramientas fáciles de usar, que soportan una gama completa de capacidades de flujo de trabajo analítico y no requieren de una intervención significativa del sector tecnológico.

En este apartado vamos a basarnos en la información obtenida a partir de algunos de los informes anuales que presenta **Gartner**.

Gartner [1] es una fuente de información fiable y con prestigio a la hora de analizar y valorar las distintas aplicaciones de inteligencia de negocios como, por ejemplo a través del *cuadrante mágico de Gartner* que veremos a continuación. Además, su trabajo de investigación y consultoría de las tecnologías de la información abarca muchos más ámbitos como, por ejemplo, analizar y mostrar la evolución comparativa de las empresas de venta de smartphones o preveer el crecimiento de la nube pública en 2017.

Vamos a ver, en concreto, dos informes de gran relevancia que **Gartner** publica anualmente:

- *"Gartner's Magic Quadrant for Business Intelligence and Analytics Platforms. 2016".*
- *"Critical Capabilities for Business Intelligence and Analytics Platforms. 2016"*: Muestra la evaluación de los productos que aparecen en el *cuadrante mágico de Gartner*.

2.2.1. Cuadrante Mágico de Gartner

Antes de pasar a ver las características de las aplicaciones más conocidas de inteligencia de negocios, vamos a presentar el **Cuadrante Mágico de Gartner**, donde se analiza la industria de las tecnologías de la información, las tendencias del mercado y se muestra la situación de los proveedores de inteligencia de negocios.

En el mercado existen infinidad de herramientas Business Intelligence. En este caso, para ver en que podemos basar la toma de decisión de que aplicación sería la más conveniente, se va a analizar un estudio que publica anualmente la consultoría Gartner, teniendo en cuenta tanto la opinión de expertos en la materia, como de los usuarios que usan los programas.

Dada la gran cantidad de plataformas de inteligencia de negocios entre las que podemos elegir el hecho de comparar todas y cada una de las posibilidades es prácticamente imposible. Para facilitar la búsqueda y la comparativa entre tantas propuestas existe el conocido *Cuadrante Mágico de Gartner*. Aunque existen otros muchos estudios (Forrester, Ventana, Barc. . .), vamos a apoyarnos en el de Gartner.

El **Cuadrante Mágico de Gartner** es un herramienta que nos permite hacer ese primer filtro. Es un estudio que ofrece un análisis cualitativo sobre el mercado y sus tendencias, teniendo en cuenta la madurez y los participantes.

El cuadrante posiciona a las plataformas de inteligencia de negocios más altas o más bajas según diferentes criterios que favorecen el conocimiento de los usuarios, el ahorro económico y la mejora competitiva de la empresa, como:

- Visión del negocio y analítica de autoservicio.
- La accesibilidad, agilidad y la capacidad analítica con datos de distintas fuentes.
- Facilidad de visualización.
- Movilidad: Poder ver los datos en cualquier sitio y plataforma.

[1] http://www.gartner.com/technology/home.jsp

Figura 2.1: Magic Quadrant for Business Intelligence and Analytics Platforms (2016)

Podemos ver, en la figura 2.1, como se encuentran situados las principales plataformas de inteligencia de negocios. Además se han señalizado las aplicaciones que vamos a analizar en este capitulo: Qlik, Tableau, Pentaho y Microsoft BI Analytics: Power BI.

2.2.2. Capacidades para plataformas Analíticas y de BI.

A través de una solución completa de Inteligencia de negocios una empresa obtiene los siguientes aspectos:

- Observar: ¿qué está ocurriendo?
- Comprender: ¿por qué ocurre?
- Predecir: ¿qué ocurriría?
- Colaborar: ¿qué debería hacer el equipo?
- Decidir: ¿qué camino se debe seguir?

Para poder alcanzar de manera correcta y fiable todos estos aspectos, una aplicación de inteligencia de negocios debe presentar las siguientes características fundamentales y críticas que *Gartner* considera:

- **Seguridad y administración**. La seguridad y gobernanza según los niveles de mando de la empresa es algo fundamental, es decir la escalabilidad empresarial. Seguridad de la plataforma en la administaración,

utilización y acceso a ella por parte de los usuarios.

- **Conectividad a las fuentes de datos**. Uno de los aspectos más importantes es que la aplicación de inteligencia de negocios sea compatible con múltiples fuentes de datos y tipos de archivo. Permitir a los usuarios conectarse a los datos contenidos en diversos tipos de plataformas de almacenamiento.
- **Cloud BI**. Capacidad para crear, implementar y administrar aplicaciones analíticas y de análisis en la nube.
- **Administración de la plataforma BI**. Capacidad para optimizar el rendimiento y garantizar una alta disponibilidad de la plataforma, así como la recuperación ante situaciones "desastre".
- **Almacenamiento de datos y herramientas ETL propias**. Acceder, integrar, transformar y cargar datos en una capa de almacenamiento autónoma, con la capacidad de incorporar y gestionar cargas de datos y actualizarlos.
- **Preparación de los datos de manera autónoma**. Autodescubrimiento semántico, combinaciones inteligentes de datos, perfilado inteligente, generación de jerarquías entre los datos, creación de modelos analíticos, etc.
- **Dirección y manejo de los MetaDatos**. Herramientas que permiten a los usuarios compartir un mismo modelo de sistema de regristro de metadatos. Esto proporciona una forma robusta y centralizada para buscar, capturar, almacenar, reutilizar y publicar objetos procedentes de los metadatos.
- **Introducir análisis avanzados**. Permite a los usuarios acceder fácilmente a capacidades avanzadas de análisis.
- **Visualización interactiva a la hora de explorar los datos**. Característica imprescindible que permite a los usuarios analizar los datos interactuando directamente con una representación visual. La exploración de datos se lleva a cabo a través de múltiples opciones como la manipulación de imágenes en los gráficos, el color, el brillo, el tamaño y la forma, el movimiento de objetos, etc. Las opciones de visualización deben de ir más allá de las gráficos de sectores, o de barras y líneas, dirigiéndose hacia unas representaciones más sofisticadas como, por ejemplo, mapas de calor, mapas geográficos, árboles, etc.
- **Cuadros de mando**. Capacidad de crear cuadros de mando altamente interactivos, con exploración visual donde se muestran análisis avanzados de manera geo-espacial.
- **Capacidad de representación en dispositivos móviles**. Desarrollo y entrega de contenido en dispositivos móviles, aprovechando las capacidades naturales de estos dispositivos: pantallas táctiles, cámaras, conocimiento de la ubicación, etc.
- **Introducir contenido analítico**. Desarrolladores y soporte para crear y modificar contenido analítico, visualizaciones y publicaciones introduciéndolos en un proceso de negocio, en una aplicación o portal.
- **Publicación de resultados y contenido de análisis**. Permite a los usuarios publicar el contenido análitico a través de diversos tipos de salida con distintos métodos de distribución, con soportes para búsqueda de contenido, narración de historias, programación y alertas.
- **Social BI (comunidad)**. Los usuarios pueden compartir información, análisis, y diferentes contenidos analíticas a través de discusiones (foros), chats y anotaciones.
- **Facilidad de uso**. Referente a la facilidad de administración e implementación, así como la de consumo de los usuarios finales.
- **Desvinculación del departamento IT**. Los requisitos de inteligencia de negocios deberían ser llevados a cabo por los propios usuarios de negocio, es decir con una gestión de autoservicio. Para que una herramienta de autoservicio de inteligencia de negocios sea realmente útil debe permitir, tanto a los usuarios ocasionales o experimentados como a los ejecutivos, realizar nuevas consultas, informes e incluso gestionar los cuadros de mando. De esta forma, ya que se reducen costes, se permite que el

servicio se ejecute en su propia plataforma y con una menor dependencia del sector informático para mantener y gestionar el sistema, reduciendo costes.

- **Actualización**. Permite actualizar en tiempo real los datos, proporcionando información a medida que se van agregando los datos.
- **Análisis de los datos**. Realización de análisis predictivos utilizando información pasada y actual con el fin de predecir informes futuros o realizar análisis Ad Hoc a partir de los cuales se puede obtener datos actuales de un informe existente (proceso útil para responder preguntas específicas).

Las distintas capacidades que se listan arriba se pueden resumir en los siguientes grandes grupos:

1. **Aprovisionamiento de BI centralizado**. Se refiere al planteamiento de inteligencia de negocios tradicional, en el que los departamentos de IT preparan soluciones para usuarios finales, cuyo análisis suele estar más o menos limitado.
2. **Analítica descentralizada**. Posibilidad, de la que disponen los usuarios finales, de realizar análisis con más o menos independencia del departamento de IT.
3. **Governed Data Discovery**. Disponibilidad de herramientas que permiten el descubrimiento de datos de una forma "gobernada", esto es dar la libertad necesaria al usuario para el análisis de datos sin perder el control que se requiere en la administración de la información.
4. **BI embebido/OEM**. Capacidad de extraer funcionalidades de BI y embeberla en distintas aplicaciones.
5. **Despliegue en extranet**. Exposición de las soluciones de BI en internet, y no solo dentro del Firewall de la empresa como se venía realizando históricamente.

2.3. QLIK

QLik [2] fue fundada en Suecia en 1993, en sus inicios, llamada Quik, acrónimo de "Quality, Understanding, Interaction, Knowledge ("Calidad, Entendimiendo, Interacción, Conocimineto").

Actualmente está situada en Radnor, Pennsylvania, con oficinas alrededor de todo el mundo, con más de 1700 socios y con cobertura en más de 100 países.

QLIK, empresa revolucionaria en el sector, es la compañía proveedora de dos aplicaciones de *inteligencia de negocios y visualización*, llamadas: *QlikView* y *QlikSense*. Actualmente, es una de las soluciones de inteligencia de negocios más populares del mercado, siendo líder en el mercado del análisis visual.

Qlik tiene una amplia y diversa cartera de clientes a lo largo del mundo, alrededor de unos 40000 clientes, desde empresas punteras o instituciones educativas hasta entidades gubernamentales y medios de comunicación. Algunos de los clientes que podemos destacar son: Albany Times Union, Austin Fire Department, Centerstone Research Institute y Global Retail Bank, entre otros.

2.3.1. Principales características

Qlik sobresale en:

- Capacidades de exploración visual interactiva.
- Satisfacción al cliente.

[2] http://www.qlik.com/es-es

- Análisis rápido y flexible.
- Exploración con menos restricciones: el motor de indexación asociativa de datos, QIX, encuentra respuesta a las preguntas que surjan.
- Amplia movilidad en dispositivos: instalación en tablet, PC, móvil o diferentes formátos.
- Plataforma análitica donde crear: los desarrolladores pueden crear de forma rápida y fácil aplicaciones web sofisticadas y personalizar las visualizaciones.
- Facilidad y amplitud de uso.
- Procesos de Data Discovery.
- Presenta un protocolo de búsqueda similar a Google para encontrar datos.

2.3.2. Qlik Sense

Qlik Sense es una plataforma de análisis de datos.

Con Qlik Sense podemos:

- Analizar datos y descubrir cosas por nosotros mismos.
- Podemos compartir los conocimientos y analizar datos en equipos o en toda la empresa.
- Nos permite formular y responder nuestras propias preguntas, así como seguir nuestro propio camino de investigación e indagación en los datos

Este producto ofrece diferentes ediciones:

- *Para Uso personal.* **Qlik Sense Deskpot** es, en este caso, una herramienta gratuita que cualquiera puede usar en su propio ordenador. Permite crear visualizaciones de datos interactivas, informes y cuadros de mando totalmente personalizados por el usuario.
- *Para Uso empresarial.* **Qlik Sense Enterprise** ofrece una amplia gama de posibilidades de integración y visualización de datos a través de aplicaciones personalizadas para que sean desarrolladas a nivel empresarial.
- *Para Alojamiento de datos.* **Qlik Sense Cloud** ofrece un alojamiento seguro de los datos, pudiendo compartir sin riesgos, solo con aquellos usuarios autorizados, las aplicaciones creadas a través de Qlik.

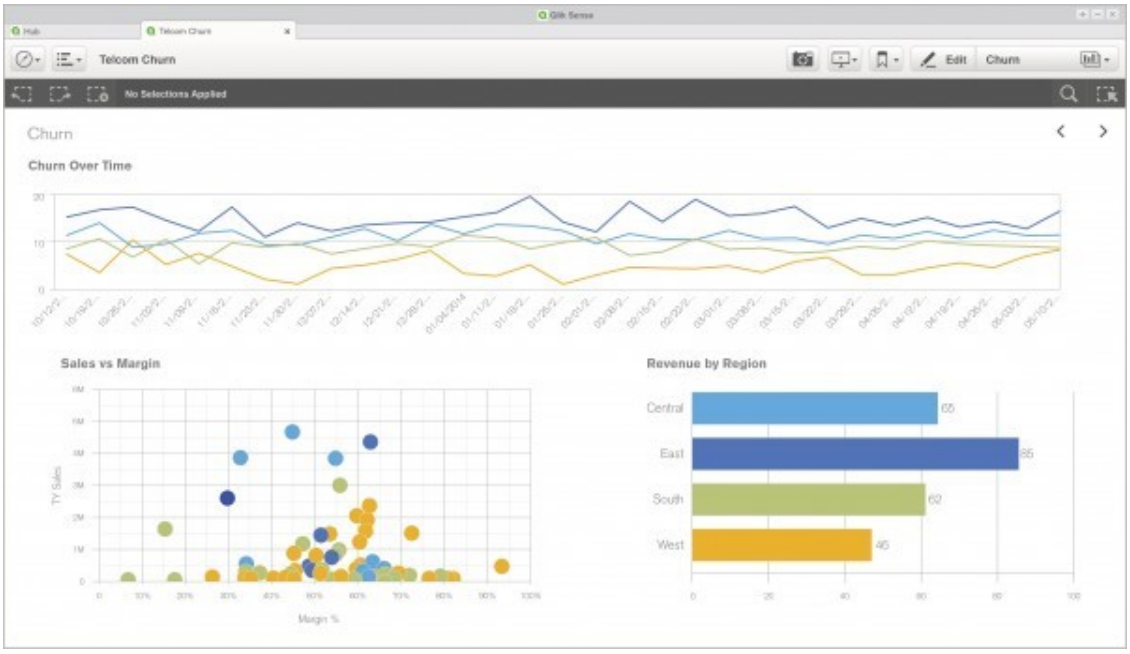

Figura 2.2: Cuadro de mandos de QlikSense

EJEMPLO Qlik Sense

A continuación, podemos ver un recorte digital del periódico *Elmundo.es*, que hace uso de aplicaciones Qlik para mostrar algunas de sus noticias a los lectores.

Figura 2.3: Ejemplo QlikSense utilizado por el periódico Elmundo.es

En la figura 2.3, podemos ver el importe de multas en las principales zonas de Madrid. De este *cuadro de mandos* podemos obtener diferente información de manera sencilla y clara:

- Zonas ordenadas de mayor a menor según la cuantía del importe de las multas.
- Localización geográfica de esas zonas representando, mediante grosor y color del circulo, las zonas donde hay mayores multas.

2.3.3. QlikView

QlikView es una poderosa herramienta de **BI** de fácil utilización para el usuario. Permite recolectar datos desde diferentes fuentes u orígenes (bases de datos SQL, datos Excel, Data Warehouse, etc) y configurarlos al gusto del usuario. Se caracteriza por un manejo fácil y una visualización llamativa y clara.

Sus principales características son las siguientes:

- Herramienta de fácil manejo para el usuario que ayuda a adquirir una visión global, unificada y coherente de la información contenida en diferentes bases de datos.
- Se basa en la búsqueda asociativa. Permite acceder a los datos, navegar e interractuar con ellos de forma intuitiva y natural, de la misma forma en la que piensan los usuarios, es decir, de forma asociativa. El usuario decide las preguntas que quiere hacer y, con un simple clic, puede seleccionar el tema del que desea recibir más información.

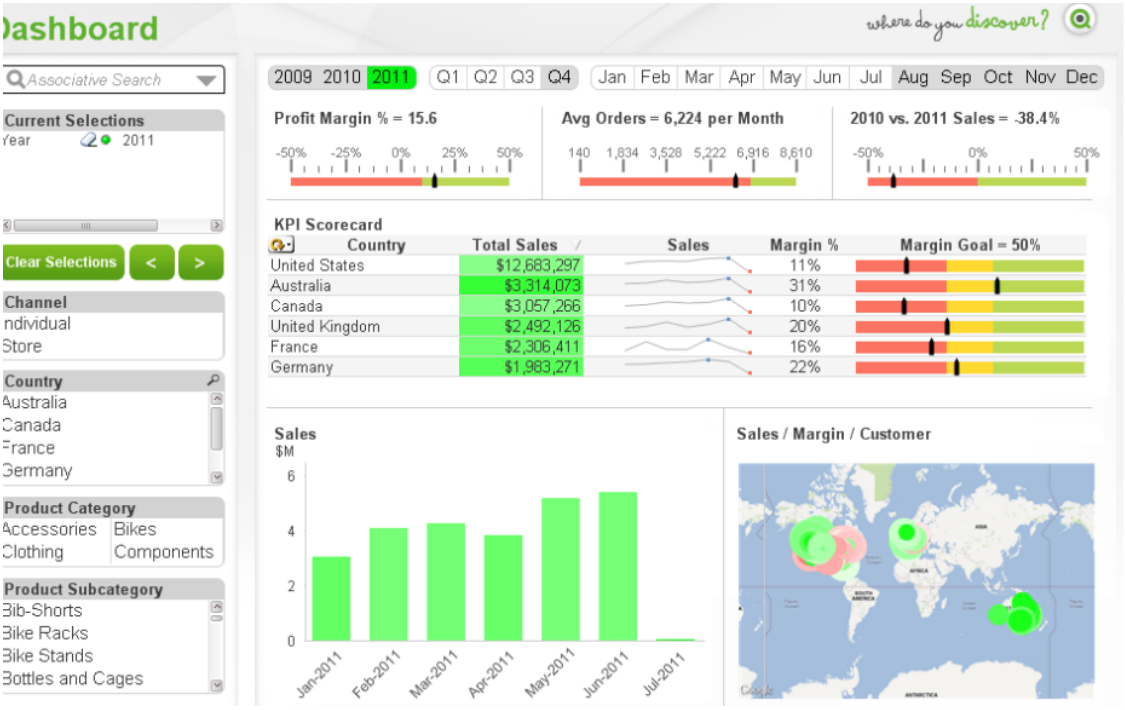

Figura 2.4: Cuadro de mandos QlikView

- Funcionamiento basado en un analítica guiada. Facilita el descubrimiento de información, la detección de conexiones y relaciones entre los datos para la correcta toma de decisiones.
- Gestión centralizada.
- Modelado de datos a través de un lenguaje de programación propio, como podemos ver en la figura 2.5, muy parecido al lenguaje SQL y de fácil comprensión.

```
Suma las ventas de las selecciones realizadas en el Estado_A, para el año 2011 y para Sucursal Norte
sum( {Estado_A  < Año={2011}, Sucursal={'Norte'} > } Venta )
```

Figura 2.5: Lenguaje de programación Qlik.

- Integración de datos con el propio cuadro de mandos, por lo que no influye los orígenes de los datos.
- Combinable con amplias aplicaciones empresariales y programas de gestión.
- Elabora presentaciones mediante tablas dinámicas y diferentes gráficas basadas en los datos. Dispone de una amplia gama de posibilidades: Velocímetros, Gráficos de barras, de Tarta, etc. Los gráficos son interactivos a través del *Drill Down*, pudiendo profundizar en el análisis y concretar en la selección de datos, para obtener información sobre aquellos aspectos que nos interesen.

- Acceder a los datos desde diferentes plataformas Windows, Mac, Linux y desde dispositivos móviles.
- Generar y distribuir informes que se pueden integrar en documentos de *Microsoft Office* a través de otro de sus productos, *Qlik NPrinting*.

EJEMPLO QlikView

Algunos de los ámbitos donde se emplean las aplicaciones QlikView son en actividades relacionadas con sistemas financieros, recursos humanos, análisis de mercado, atención al cliente, controles de la producción, etc.

Como podemos ver en la figura 2.6, tenemos un ejemplo de la creación de una aplicación QlikView a través de un fichero de datos Excel.

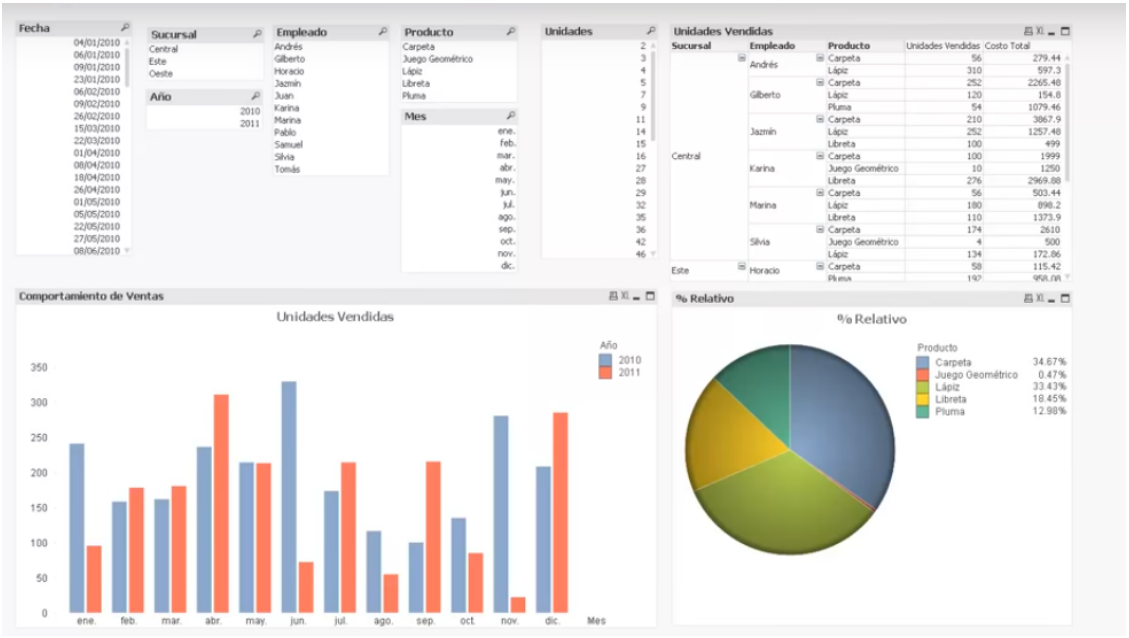

Figura 2.6: Cuadro de mandos de QlikView

En los gráficos de la figura 2.6, vemos información relativa al número de unidades vendidas por meses para los años 2010 y 2011. Por otro lado también nos muestra un diagrama de sectores donde se muestra la participación de cada tipo de productos en la ventas, donde podemos ver que las carpetas es lo que más se vende.

Toda la información que aparece en la parte superior en ventanas son posibles filtros que se pueden seleccionar. Por ejemplo, podemos filtrar para el mes de enero o por las ventas por empleado.

2.3.4. Qlik Sense o QlikView

En este apartado hemos podido ver que ambos productos tienen bastantes similitudes y puntos en común, sin embargo están orientados a fines distintos:

- QlikView, como hemos visto en el apartado anterior, es una herramienta para aquellas empresas que busquen tener aplicaciones de analítica guiada. Esto hace referencia a que esas aplicaciones están *preconfiguradas*, es decir, desarrolladas por programadores que han dedicado tiempo a armar los diferentes modelos de datos, el diseño, los gráficos, etc. El consumidor final no tiene restricción alguna para navegar entre la información: puede explorar los datos, seleccionarlos, “hacer zoom” en detalles de interés, etc, lo que le permitirá descubrir información relevante y encontrar respuestas a sus intereses.

Sin embargo, el usuario si encuentra limitaciones a la hora de generar nuevas o diferentes visualizaciones. Está pensado para situaciones en las que el usuario solo consume información, como entornos de oficina.

- Qlik Sense se centra en ofrecer autonomía y autoservicio al usuario mucho más fácil de usar que QlikView. Pensado para un entorno móvil y táctil donde no se requiere de tanta preconfiguración previa si no que, todo lo contrario, se busca ofrecer al usuario la flexibilidad de construir sus propias visualizaciones, cambiar los diseños, etc. Esto permite al cliente estar mucho más involucrado en el proceso y el resultado final. Este producto es el idóneo si hablamos de usuarios nuevos o de aquellos que deseen una mayor interacción de autoservicio en la exploración de los datos.

2.4. Tableau

Tableau [3] Software ofrece productos de inteligencia de negocios y análisis de datos. Fue fundada en 2003 por los pioneros Chris Stolte, Pat Hanrahan y Christian Chabot, con sede en Seatle.

Tableau es una potente herramienta de cuadros de mandos y análisis, fácil de usar y con la última tecnología. Permite conectarse a los datos y convertirlos en información. Descubrir, analizar e identificar tendencias en segundos, publicar cuadros de mandos y compartirlos dentro de la empresa.

Presentan diferentes productos:

- **Tableau Desktop**: Preparado para un usuario medio. Es muy rápido y fácil de utilizar, presenta cuadros de mandos brillantes, conexión directa y sin programación, mashups perfectos, etc.
- **Tableau Server**: Especial para empresas. Presenta portales web interactivos, no requiere de desarollos informáticos, acceso veloz a millones de datos, mezcla diferentes datos con un solo click y permite utilizar y manipular los datos en tiempo real en las reuniones.
- **Tableau Digital**: Para integración en *sides* de noticias e información. Mejora la presentación de la página con datos interactivos y actualizados. Posibilita un servicio personalizado que ofrece una mayor repercusión ya que el sistema estará conectado con las redes sociales como Facebook.
- **Tableau Public**: Para integración con blogs. Es un servicio gratuito donde no es necesario ser un programador. Permite que los usuario exploren los datos en línea y el usuario puede compartir sus propias visualizaciones.

Con más de 50000 clientes, Tableau ofrece servicio a una amplia variedad de tipos de compañías, incluyendo grandes corporaciones, bancos, instituciones educativas y médicas. Tiene una larga trayectoria y, durante su desarollo, ha participado en múltiples procesos de inteligencia de negocios. Destacan entre ellas: The Coca-Cola Company, Heineken, Johns Hopkins University, Linkedin y Yahoo.

2.4.1. Principales características

Las principales características que diferencian a Tableau son las siguientes:

- **Acceso instantáneo a los datos**: Independientemente de los orígenes de datos que tenga la empresa, Tableau puede gestionarlos (Excel, bases de datos y archivos propios). El motor de Tableau le permite trabajar con múltiples fuentes de datos a gran velocidad
- **El propio usuario gestiona y distribuye sus análisis**: Normalmente, al departamento de infromática se le solicita hacer un análisis desde otro departamento. Esto conlleva a errores y a que la realización de los informes se alarguen en el tiempo. El usuario puede enviar y compartir sus análisis

[3] https://www.tableau.com/es-es

a través de diferentes versiones que ofrece Tableau: Versiones Reader(gratuito) y Server (web).Esto reduce su dependencia del departamento de IT a la hora de realizar análisis.

- Conexión online y offline a los datos de la empresa.
- **Acceso inmediato a los datos**: Información stock por ubicación e información de inventarios disponible a cualquier fecha.
- **Gran capacidad visual de análisis**.

  Como podemos ver en la figura 2.7, ofrecen visualizaciones agradables, sencillas y productivas intentando lograr que los datos sean comprensibles para cualquier persona, sin necesidad de conocimientos analíticos o técnicos.

  Ofrece otras posibilidades como visualización de la información de manera territorial, como vemos en la figura 2.8.

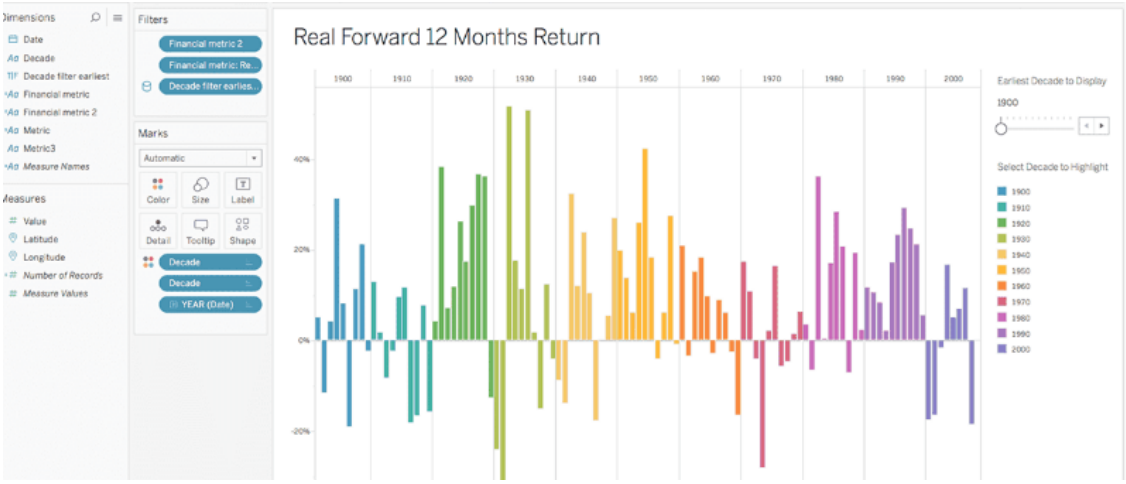

Figura 2.7: Ejemplo Cuadro de Mandos generado por Tableau

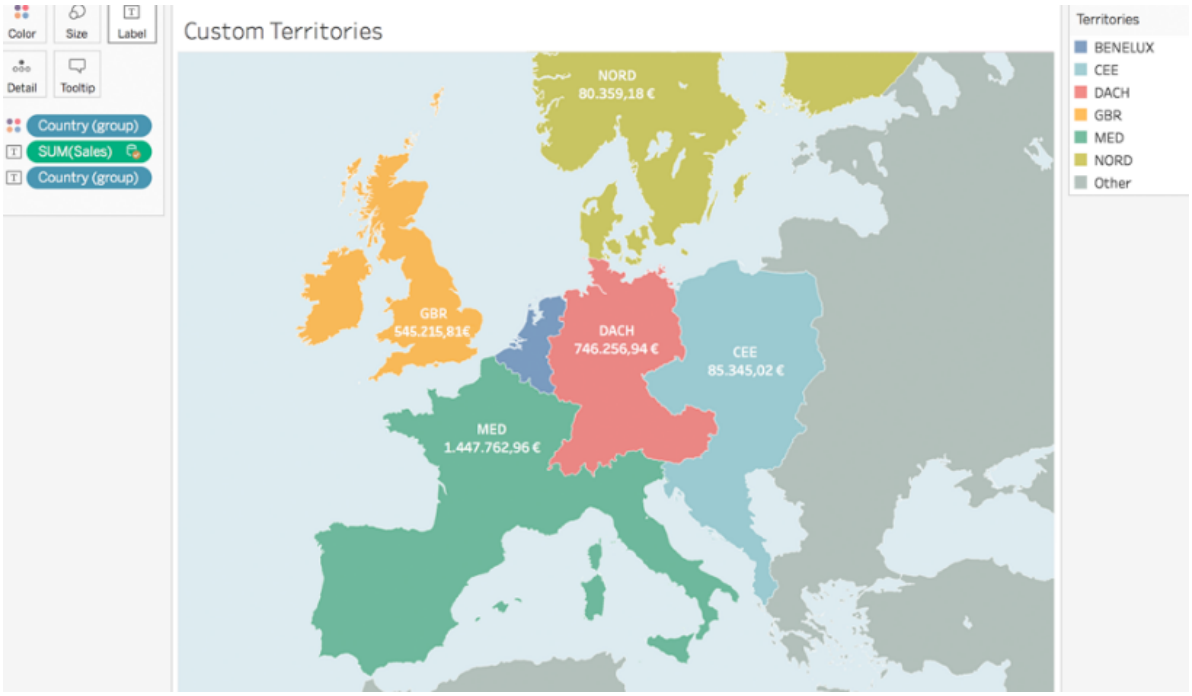

Figura 2.8: Cuadro de mandos sobre mapa. Tableau

- **Es intuitivo**. Se aprende a utilizarlo muy rápidamente y es de fácil uso.
- **Precio**. Su precio está por debajo de la media del mercado.
- Permite análisis más profundos de manera rápida, eficaz y sencilla. Principalmente obtener análisis de tendencias, regresiones (figura 2.9), correlaciones simplemente arrastrando y clicando en el *cuadro de mandos*

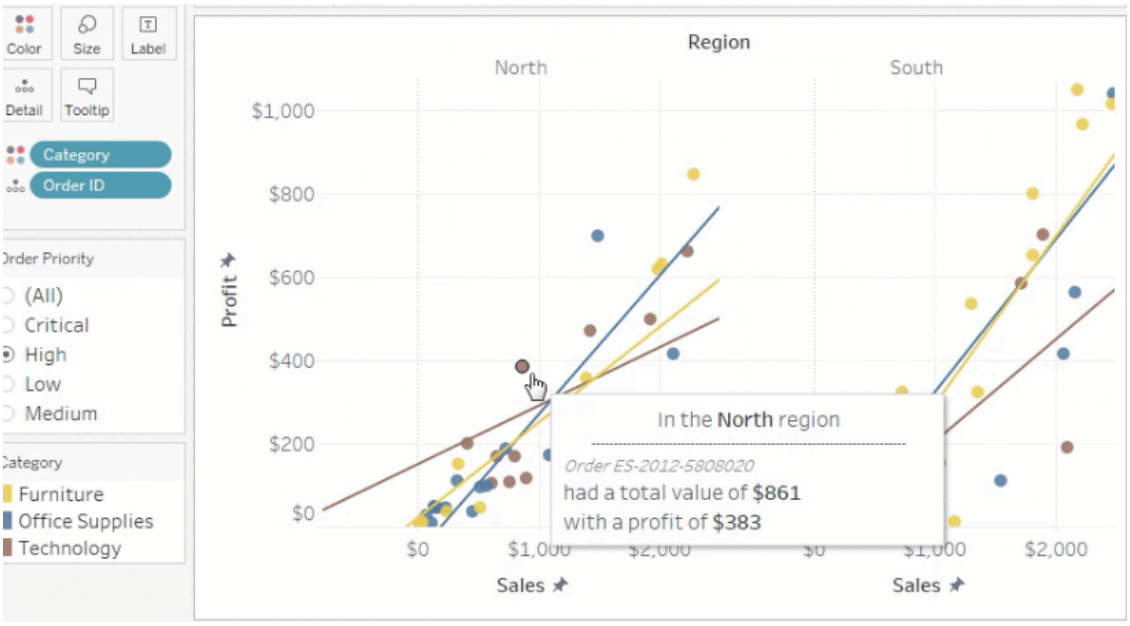

Figura 2.9: Cuadro de mandos con regresiones. Tableau

- Elaborar predicciones sobre lo que puede suceder.

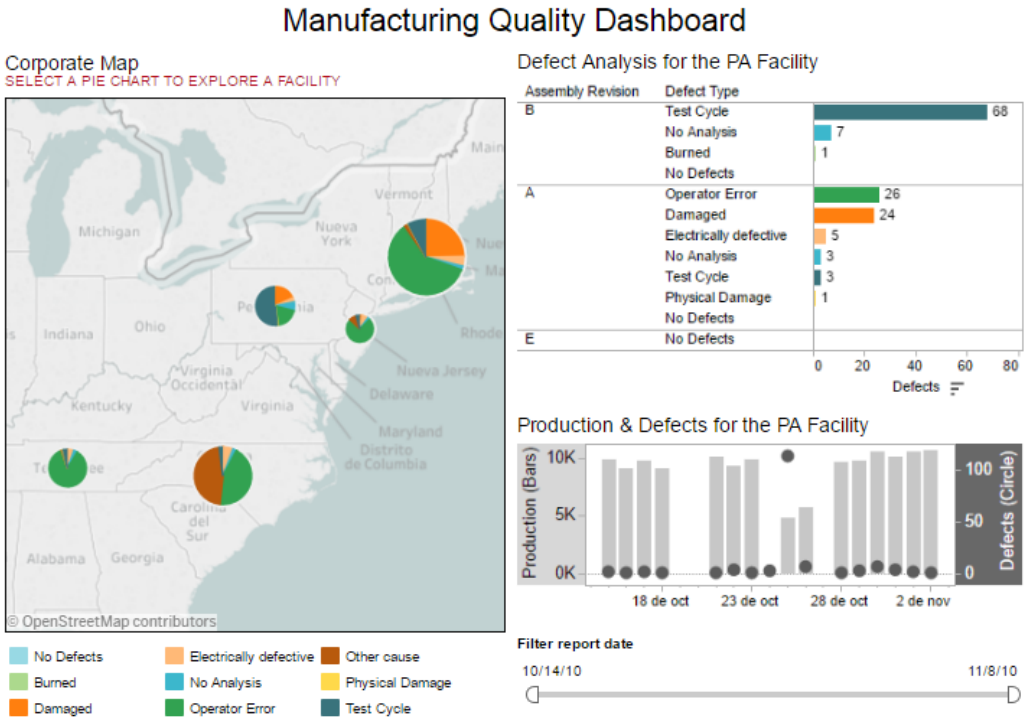

Figura 2.10: Ejemplo de Cuadro de mandos con Tableau.

2.5. Pentaho

Pentaho [4] es una herramienta de inteligencia de negocios desarrollada bajo la filosofía *"open source"* y sin costes de licencia para la gestión y toma de decisiones empresariales.

Es una plataforma compuesta de diferentes programas que satisfacen los requisitos de la inteligencia de negocios, ofreciendo soluciones para la gestión y análisis de la información.

Pentaho nació de la voluntad de lograr un cambio positivo en el mercado de análisis de negocios, dominado por mega proveedores que ofrecían sistemas de BI costosos y pesados productos basados en plataformas tecnológicas obsoletas. Cinco veteranos de la industria buscaron una solución a este problema (de ahí el nombre: "Penta").

Pentaho ha sido pionera en el futuro de la inteligencia empresarial y de análisis disponiendo un conjunto completo de capacidades para el acceso e integración de datos, descubrimiento, análisis y visualización.

La compañía tiene su sede en Orlando, Florida, con oficinas en San Francisco, California y en toda Europa.

En 2015, *Hitachi Data Systems* adquirió a la compañia como parte de su proyecto de innovación. Hitachi es una empresa lider en "Internet of Things" asi como en el análisis de Big Data. Pentaho conserva su nombre y continua operando de forma independiente.

[4] http://www.pentaho.com/

Algunas de las compañias que utilizan Pentaho son, entre otras, Telefónica. Logitech o The New York Times.

2.5.1. Principales características.

La solución Business Intelligence OpenSource Pentaho pretende ser una alternativa a las soluciones propietarias tradicionales más completas: Business Objects, Cognos, Microstrategy, Microsoft, etc... por lo que incluye todos aquellos componentes que nos podemos encontrar en las soluciones BI propietarias más avanzadas.

El siguiente esquema nos muestra la arquitectura estructurada de las diferentes componentes que forman parte de Pentaho, en el que cabe destacar el poderoso motor de Data Mining del que disponen.

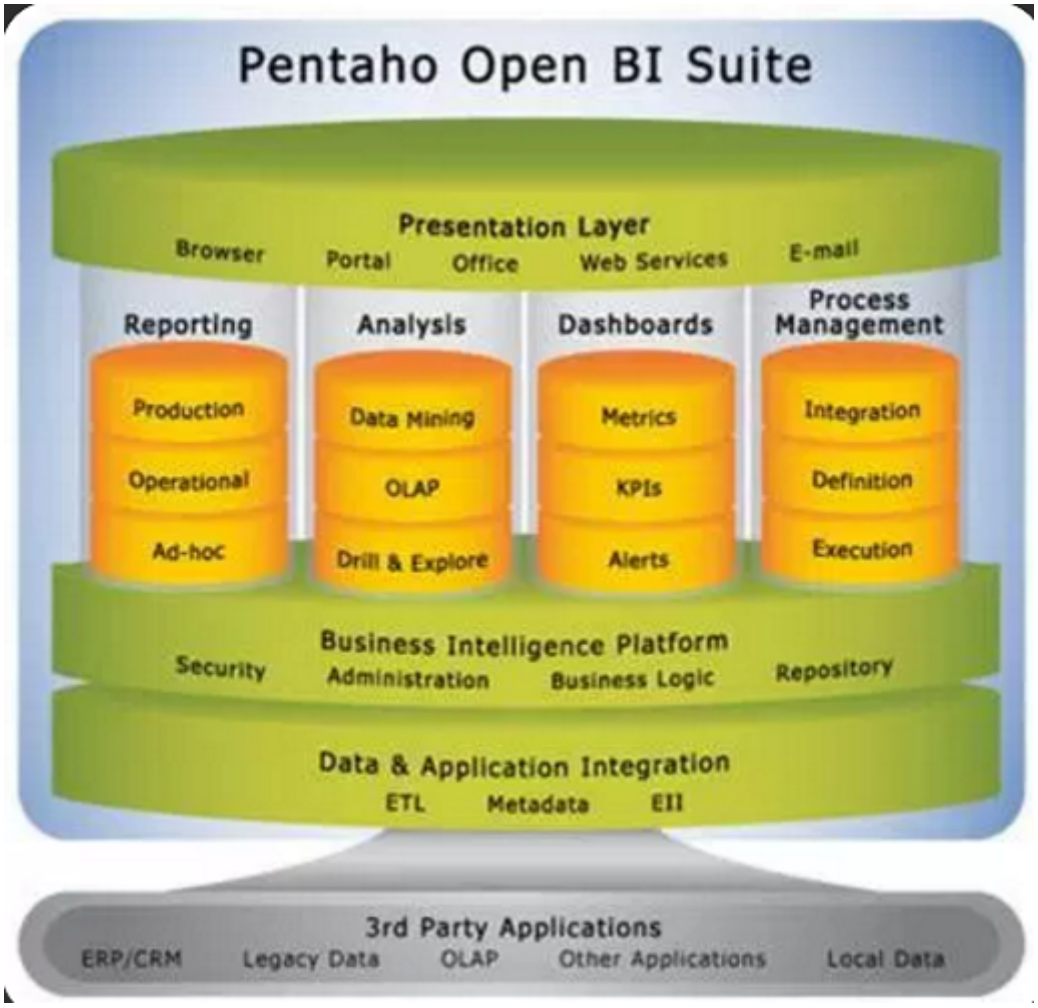

Figura 2.11: Esquema Productos Pentaho

Entre sus productos destacan: Pentaho Business Analytics, Pentaho Data Integration, que suele venir integrado en el primero, y Pentaho Reporting .

Pentaho Data Integration

Muchas organizaciones tienen información disponible en aplicaciones y bases de datos separadas. **Pentaho Data Integration** abre, limpia e integra esta valiosa información y la pone en manos del usuario.

- Provee consistencia. Una sola versión de todos los recursos de información, que es uno de los más grandes desafíos de hoy en día para las organizaciones IT .
- Dispone de una poderosa ETL (Extracción, Transformación y Carga).
- El uso de kettle permite evitar grandes cargas de trabajo de manera manual, que son frecuentemente difícil de mantener y de desplegar.

- Usa tecnologías estándar: Java, XML, JavaScript.
- Multiplataforma: Windows, Macintosh, Linux.
- Entorno gráfico de desarollo.

Dispone de una amplia gama de análisis que podemos aplicar a los datos, desde sencillos informes descriptivos, hasta modelización de modelos predictivos.

Pentaho Reporting

Módulo incorporado en Pentaho que permite:

- Generar informes ágiles y de gran capacidad.
- La distribución de los resultados se ofrece en múltiples formatos: opción de imprimir o exportar a formato PDF, XLS, HTML y texto.
- Permite programación de tareas y ejecución automática de informes con una determinada periodicidad.

Pentaho Bussines Analytics

Herramienta que, mediante una interfaz de tipo web, permite analizar y generar diferentes tipos de visualizaciones interactivas, así como la generación de informes y análisis predictivo. Las características más destacables son las siguientes:

- Descubrir patrones ocultos y correlaciones en los datos.
- Prevenir errores futuros basados en patrones históricos, como los que vemos en la figura 2.12.

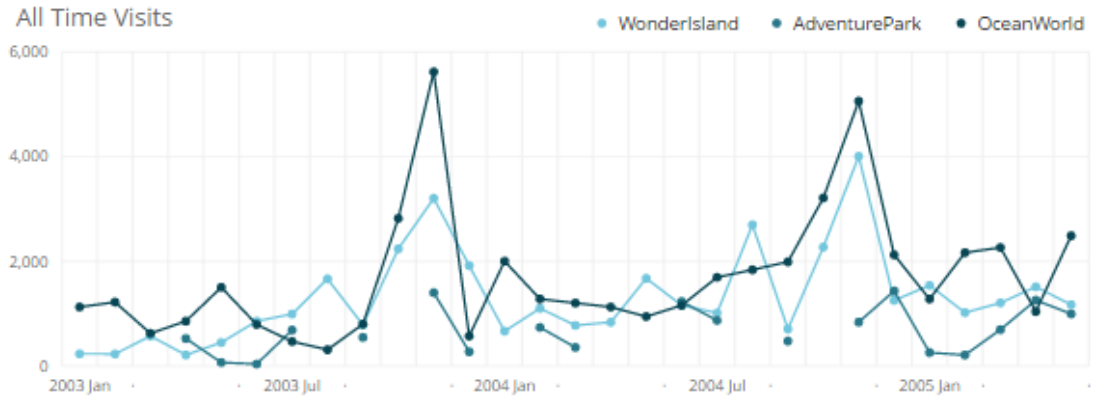

Figura 2.12: Tendencia de las visitas de parque de atracciones. Pentaho

- Cuadros de mandos interactivos sobre los análisis: Filtrar, hacer “zoom” a los datos, resaltar información, etc.

- Dispone de gráficos de localizaciones, de burbujas, como vemos en la figura 2.13, entre otros.

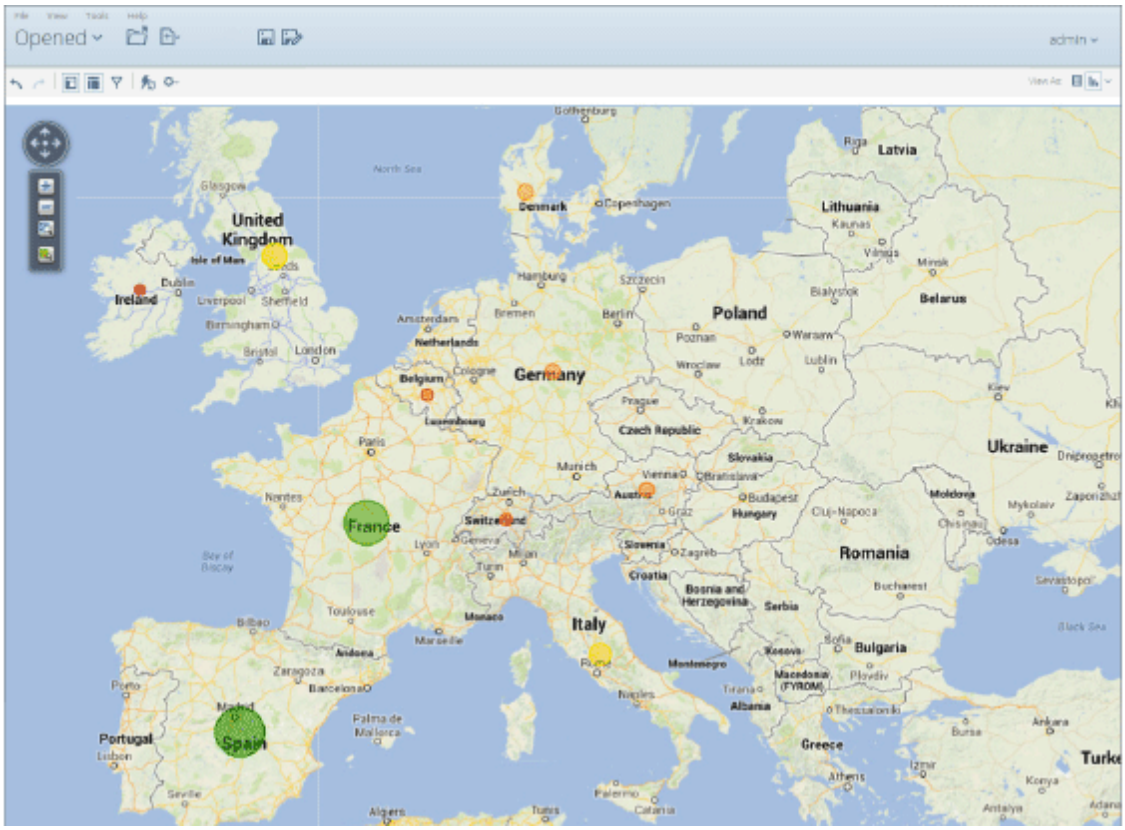

Figura 2.13: Ejemplo de Cuadro de mandos geográfico con Pentaho.

EJEMPLO Pentaho

En la figura 2.14, podemos ver un ejemplo de un cuadro de mandos en el que se muestra el tipo de crímenes más comunes en Chicago, en el espacio temporal deseado, mostrándose también la tendencia.

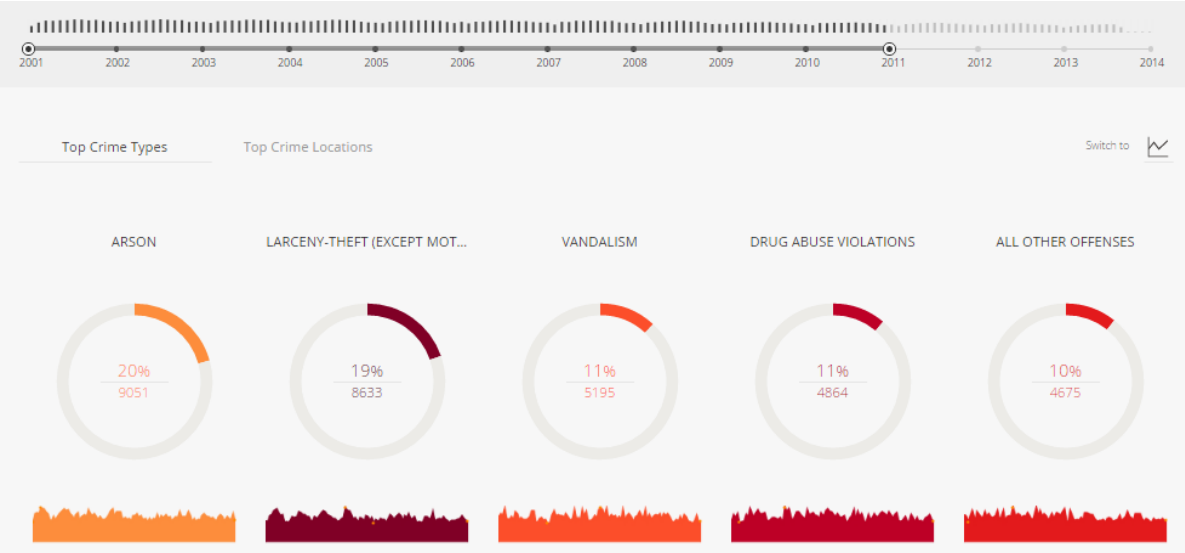

Figura 2.14: Cuadro de mandos, generado con Pentaho, sobre los tipos de crímenes más usuales en Chicago.

En concreto, estamos visualizando los crímenes desde 2001 a 2011. Vemos que en esos diez años el 20 % de los crímenes ocurridos en Chicago fueron incendios provocados.

También podemos ver, a través de la figura 2.15, en que lugares se han producido la mayoría de los crímenes.

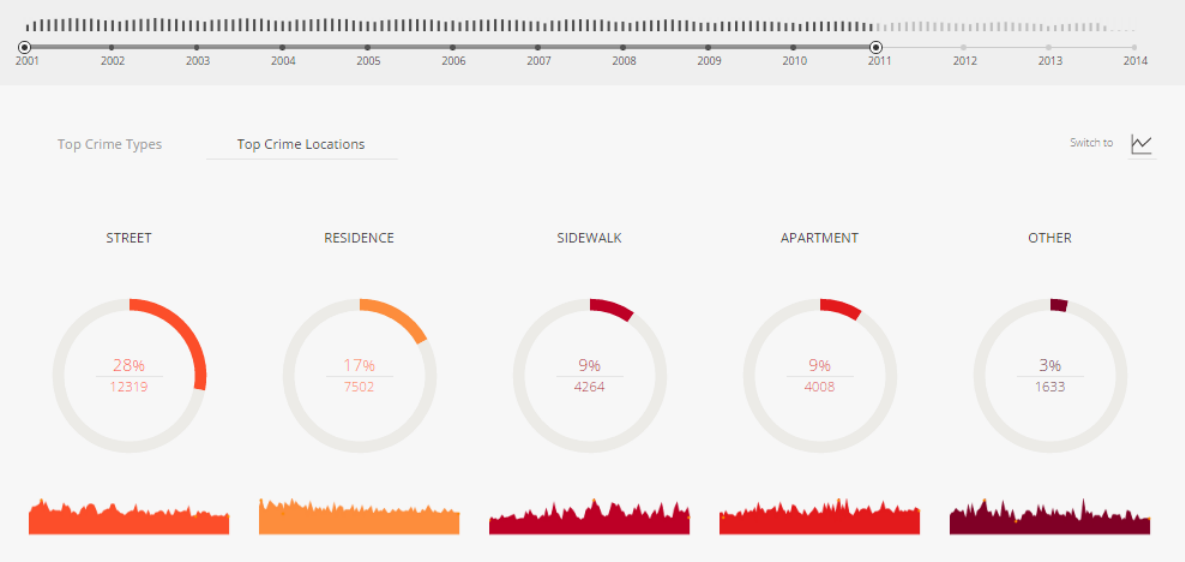

Figura 2.15: Cuadro de mandos, generado con Pentaho, sobre los lugares donde se cometen más crímenes en Chicago.

Vemos que el 28 % se han producido en las calles.

Toda esta información puede ser de gran utilidad, por ejemplo, para la planificación de estrategias en la policía para prevenir los delitos: más cuerpos de policía en las calles, campañas de concienciación de los efectos de los incendios, etc.

2.6. Microsoft: POWER BI

Microsoft presenta dentro de su oferta de BI una EBIS, Employee Benefits Information System, que contiene las propuestas tradicionales de la organización:

- SQL Server: Conjunto de herramientas que posibilitan llevar todo el proceso de BI.
- SharePoint: Herramienta colaborativa y de publicación.
- Excel & Power Pivot: Herramientas que admiten el acceso a las bases de datos para consultas y exploraciones.

Microsoft ofrece un conjunto de soluciones donde se puede identificar por un lado la solución que generaría las BBDD del proceso BI y, por otro lado, otras soluciones más relacionadas con los informes y visualizaciones que se puedan realizar sobre los datos.

Algunos usuarios notables de **PowerBI** [5] son: The Weather Channel, Volvo car Corporation y San Diego County Sheriff's Department.

2.6.1. Principales Características

Es una herramienta de ayuda a la inteligencia de negocios que se basa en la configuración rápida de paneles sencillos con los que obtener información de los datos en muy poco tiempo.

- Precio muy bajo, incluyendo versiones gratuitas.
- Muy fácil de usar.
- Integración con otros productos Microsoft, los cuales son muy utilizados en muchas empresas. Esto, presenta la contrapartida de que el verdadero rendimiento de la aplicación se alcance si se conecta con herramientas Microsoft.
- Está incrementando sus integraciones y su red de búsqueda de patrones.
- Capacidad de conectarse con una amplia gama de base de datos.
- Solo está disponible en *la nube*.
- No es la solución óptima si se quiere trabajar con bases de datos extremadamente grandes.
- Obtener una visión completa, de toda la empresa, en un único panel.
- Facilita la creación de informes finales donde plasmar la información y compartirla de forma interactiva.

SQL Server (Procesos BI)

Contiene como características principales:

- La integración de los datos a través de procesos ETL permite conectarse e integrarse con cualquier tipo de dato, como hojas de cálculo de Excel, orígenes de datos locales, conjuntos de datos de Hadoop, datos de streaming y servicios en la nube (Github, Google Analytics, Salesforce, etc).
- La generación de la base de datos analítica OLAP, mediante Analysis Services.
- Presenta funcionales de Data Mining y orientación al Big Data.

[5] https://powerbi.microsoft.com/es-es/

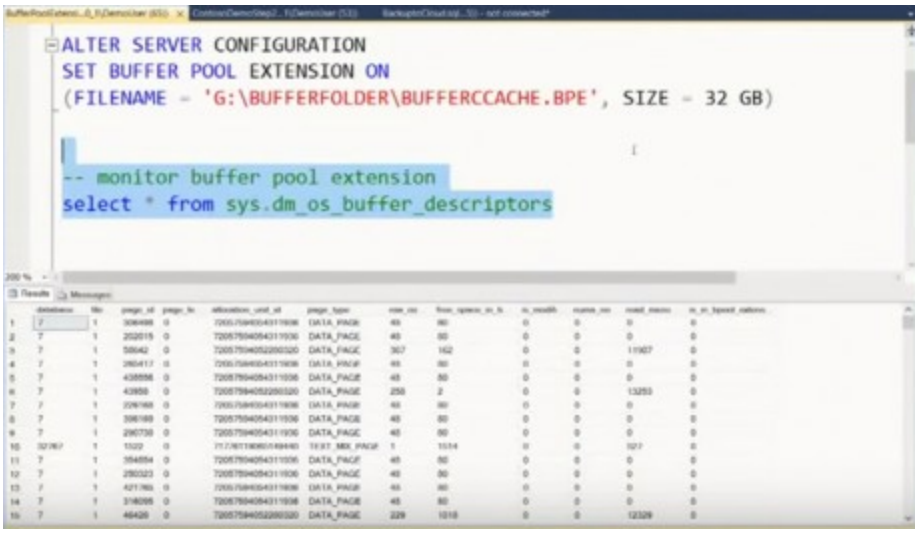

Figura 2.16: Consola SQL Server de Microsoft: Power BI

SharePoint (Publicación)

- Propone un espacio colaborativo y de publicación.
- Generación de paneles y cuadros de mando.

Excel y Power Pivot (Publicación)

- Facilita la generación de informes y gráficos.

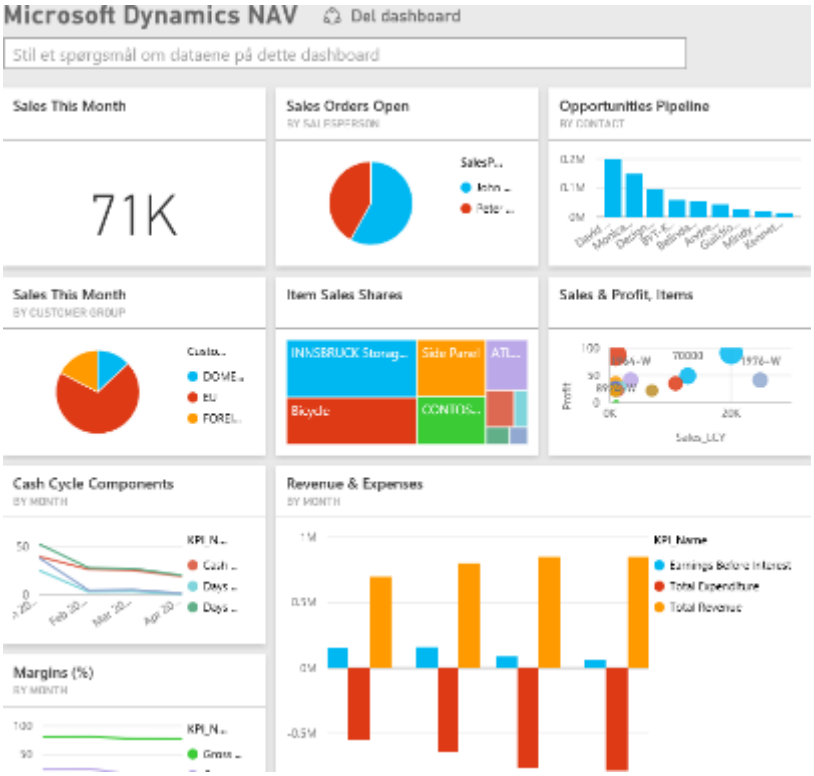

Figura 2.17: Integración con Excel y Power Pivot.

- Exploración dinámica de los datos.

2.7. Tabla Resumen de Valoración de las Aplicaciones

En base a todo lo que hemos visto en los apartados anteriores, y respecto a la información obtenida por las publicaciones de Gartner ya mencionadas, obtenemos nuestro Ranking de valoración basado en esas capacidades críticas sobre los productos y los servicios que presentan.

	Qlik	Tableau	Pentaho	Microsoft
Seguridad y Administración del usuario	3,5	4	4	3
Conectividad a la fuente de Datos	2,5	4	4	4,5
Cloud BI	1,5	2,5	1,5	4
Administración de la plataforma BI	3,5	3,5	5	4,5
Almacenamiento de datos y herramientas ETL	4,5	3	4	4
Preparación de datos automática	3	2,5	2	3
Manejo y dirección de MetaDatos	2	2	3	2,5
Introducir Análisis Avanzados	1,5	1,5	4	1,5
Interactividad Visual(Interfaz)	3,5	4	3	2,5
Dashboards	3,5	3	3	3
Dispositivos móviles	2	3	2	3
Introducir contenido analítico	4	3	4,5	3,5
Publicación de contenido y resultados	2	2	2	1,5
Comunidad BI	1,5	1,5	1	1,5
Facilidad de uso	3	3,5	2	3,5
Valoración Media	2,8	2,9	3	3

Figura 2.18: Ranking de las aplicaciones basadas en la opinión y valoración de los expertos analistas de las capacidades críticas definidas por Gartner.

Como podemos ver en la figura 2.18, tenemos las capacidades consideradas, que ya hemos mencionado, valoradas mediante una nota numérica entre 0 y 5, siendo 5 la mejor valoración posible. Dichas capacidades están valoradas para las cuatro aplicaciones de inteligencia de negocios que hemos estudiado en este capítulo.

En la figura 2.18, vemos como, para cada capacidad, se muestra en verde los valores correspondientes a los programas que presenten mejor valoración, y en rojo los peor valorados.

De manera que, dependiendo de las necesidades del usuario, a través de esta tabla resumen el usuario puede decantarse por una aplicación u otra según lo que quiera potenciar. Por ejemplo, si estamos interesados en una interfaz llamativa e interactiva vemos que la mayor puntuación la obtiene *Tableau*.

Realizando la suma de todas las capacidades, y la mejor puntuada ha sido Microsoft (45.5 puntos), seguida muy de cerca por Pentaho (45 puntos) y Tableau (43 puntos), y en último lugar se encuentra Qlik (41.5 puntos).

2.7.1. Microsoft Power BI

Es el software mejor valorado en la conectividad a la fuente de datos, su uso en la nube y dispositivos móviles así como en facilidad de uso integridad de la visión.

Como desventajas se encuentran la incapacidad de soportar una alta cantidad de usuarios y la necesidad de mejorar en la aplicación de análisis más avanzados.

Microsoft destaca en la calidad de la ayuda al usuario.

2.7.2. Pentaho

Según el informe de Gartner, Pentaho ofrece la plataforma más completa de BI Open Source, con altas capacidades de integración de datos de diferentes fuentes, así como el soporte para análisis avanzado.

Pentaho es elegido por las empresas por presentar una licencia a costes mucho más competitivos que el resto de proveedores del mercado, por su funcionalidad, por el acceso a los datos y sus capacidades de integración. En general, Pentaho se sitúa por encima de la media en muchos aspectos individuales en sus productos, destacando las herramientas de desarrollo.

Según el estudio, el futuro de Pentaho es prometedor, aunque debe mejorar para llegar a posicionarse como líder de mercado, a pesar de eso es una buena opción a bajo costo.

2.7.3. Tableau

Líder en el mercado en cuanto a su ejecución e interfaz. Es de fácil implementación y se adapta a bases de clientes más grandes, resolviendo escenarios más complejos.

Ofrece un producto muy interactivo, con disponibilidad de conectarse a diferentes tipologías de bases de datos, aunque necesita mejoras en integrarse con otras herramientas.

2.7.4. Qlik

Destaca por su almacenamiento de datos y herramientas ETL, así como en las posibilidades a la hora de crear cuadros de mandos

Debe mejorar en aspectos de conectividad a las fuentes de datos, su gestión en la nube y la aplicación de análisis avanzados.

Qlik es de los mejores puntuados en la experiencia de venta y la facilidad de uso.

Capítulo 3

R para la Inteligencia de Negocios

En los apartados anteriores, hemos presentado las principales aplicaciones comerciales que disponen de soluciones para la **inteligencia de negocios**. Vamos a introducir a **R** [1] desde la perspectiva de una herramienta de software libre para la inteligencia de negocios.

3.1. ¿Por qué R?

R es una plataforma abierta que presenta un lenguaje de programación para la estadística y permite a los usuarios *publicar* sus códigos de forma abierta para analizar y visualizar conjuntos de datos.

R fue inicialmente diseñado por una comunidad de académicos que fue creciendo a pasos agigantados hasta ser usado comercialmente por un amplio abanico de sectores. De esta forma, R, ha presentado uno de los crecimientos más rápidos, referentes al desarrollo de lenguajes de programación del mundo, posiblemente debido a su repositorio de código público, donde usuarios de todo el mundo contribuyen con sus paquetes, y a su distribución bajo licencia pública.

3.2. Aportación de RStudio

El rápido avance de R lo ha experimentado también el entorno de desarrollo **RStudio** [2] para **R**. **RStudio** avanza rápidamente a la par de las necesidades del usuario de **R**. Ofrece productos y crea paquetes que facilitan y cubren los requisitos sobre manejo de datos que se demandan en la sociedad tecnológica actual.

RStudio nos facilita tanto la tarea de uso interactivo de **R**, como la programación de código (*scripts*), a través de una interfaz amable e intuitiva.

RStudio es uno de los entornos más populares para crear aplicaciones a través del lenguaje **R**, teniendo en cuenta, además, la disposición de una versión gratuita, *open source* y multiplataforma de su programa en forma *desktop*.

Los propios miembros de **RStudio** ocupan un lugar muy destacado en la comunidad que se encarga del desarrollo en **R**. A través de esta comunidad se han creado algunas de las librerías más usadas que ofrecen funcionalidades muy útiles al programa base de **R**.

De los diferentes desarrollos que **RStudio** lidera, uno de los más ambiciosos y que presenta un gran potencial es **Shiny**.

[1] https://www.r-project.org/
[2] https://www.rstudio.com/

3.3. Aspectos básicos de las Aplicaciones de Inteligencia de Negocios

En este apartado, vamos a ver como **R** está altamente capacitado para ser una opción a las aplicaciones comerciales que nos ofrece una *"ingeniería"* que permite realizar los procesos que requiere la **BI**: Extracción de información útil y su visualización (a través del entorno de desarollo **RStudio**).

R lleva desarrollando, desde hace décadas, una amplia gama de herramientas estadísticas que están a disposición del usuario a través de cientos de paquetes que se encuentran en los múltiples repositorios existentes.

Para tener una primera idea, vamos a presentar las herramientas que dispone **R** en las acciones fundamentales de la **BI**: Extracción de los datos, su transformación, la carga de datos y su presentación.

3.3.1. Extracción y carga de datos

R presenta algunos paquetes útiles para estas acciones principales, donde podemos destacar:

- **DBI**: Interfaz para la comunicación entre R y sistemas de gestión de bases de datos relacionales.
- **RODBC**: Interfaz para conectarse con bases de datos ODBC ("Open Data Base Connectivity", estándar de acceso a las bases de datos).
- **RJDBC**: Permite a R conectarse a cualquier DBMS ("Data Base Management System", *Sistema de Gestión de Base de Datos*) que disponga de un controlador JDBC ("Java Database Connectivity").
- Función *fread* de **data.table**: Especialmente para leer de forma rápida y conveniente un conjunto de datos pesado.
- Otros muchos paquetes que soportan la lectura de datos desde diferentes formatos (xlsx, sas, spss. . .)

3.3.2. Transformación de datos

Destacamos:

- **data.table**: Presenta unas potentes herramientas para transformar los datos con una sintaxis sencilla e intuitiva.
- **dplyr**: También disponen de potentes herramientas para el manejo de datos. Esta librería presenta una sintaxis legible y fácil de entender para este fin.

3.3.3. Presentación de datos

En los dos apartados anteriores, hemos puesto el punto de mira en los procesos denominados **ETL** ("Extract, Transform, Load"), que se refieren a los procesos más técnicos.

La presentación de la *información útil* es un aspecto totalmente diferente a los procesos **ETL**. A través de R podemos crear interesantes cuadros de mandos, lanzados como aplicaciones web, sin necesidad de preparar los datos para aplicarlos a una herramienta de presentación externa.

De esta manera, se pueden alojar aplicaciones web, donde generar gráficos y realizar consultas sobre los datos de manera interactiva.

Todo esto lo realizamos a través de una sesión de **R**, es decir que tenemos a nuestro alcance el poder de ejecutar las funciones provistas en los paquetes, junto con la amplísima gama de representaciones gráficas y herramientas de análisis estadístico de las que dispone.

Las principales herramientas que nos presenta **R** son las siguientes:

- **Librería Shiny** [3]: Permite crear fácilmente aplicaciones web interactivas y dinámicas con R. Se pueden obtener aplicaciones visualmente muy llamativas y muy potentes con un mínimo esfuerzo.
- **Librería Flexdashboard** [4]: Permite crear cuadros de mandos interactivos usando **RMarkdown**.
- **Librería htmlwidgets** [5]: Incorpora a R la posibilidad de obtener visualizaciones llamativas a través de *JavaScript*.

El tema sobre la visualización de los datos es donde vamos a centrar nuestro estudio, por lo que vamos a ver más detalladamente estas diferentes posibilidades

3.4. Librería Shiny

Shiny es un producto ofrecido por **RStudio** desarrollado en 2012. Está disponible en *CRAN* a través del paquete *shiny*.

Se puede decir que, a la hora de utilizar **R**, sus desarrollos y aplicaciones son poco exportables a los usuarios finales. Para paliar este inconveniente se han usado lenguajes como *Java*, *C++* o *C*, entre otros, lo cual requiere tener un conocimiento previo para programar con ellos.

RStudio, conocido por su famoso entorno de desarrollo hacia el usuario para **R**, desarrolló este paquete, **Shiny**, para construir aplicaciones web utilizando **R**, con la gran ventaja de que no hace falta conocimiento de HTML o JavaScript, sólo conocer R.

Es una herramienta que nos permite crear, sin mucha dificultad, aplicaciones web interactivas (*apps*), que permiten a los usuarios interactuar en la visualización de sus datos de manera dinámica y sin tener que hacer ningún tipo de manipulación o cambio en el código.

La base de **Shiny** está fundamentada en la **programación reactiva** que vincula, de forma dinámica, los valores de entrada con los de salida.

Permite una interacción e incorporación de otros lenguajes de programación, como *HMTL5/CSS3*, *JavaScript+*, etc, con la presencia de objetos adaptados que "imitan" a los clásicos elementos HTML (botones, cuadros de texto, barras de desplazamiento,. . .) que se conectan con las potentes gráficas de R y su estructura de datos.

Además, dispone de **widgets** preconstruidos, como podemos ver en la figura 3.1, que permiten una construcción de aplicaciones de una visualización agradable e interactiva.

[3] https://shiny.rstudio.com/
[4] http://rmarkdown.rstudio.com/flexdashboard/
[5] http://www.htmlwidgets.org/

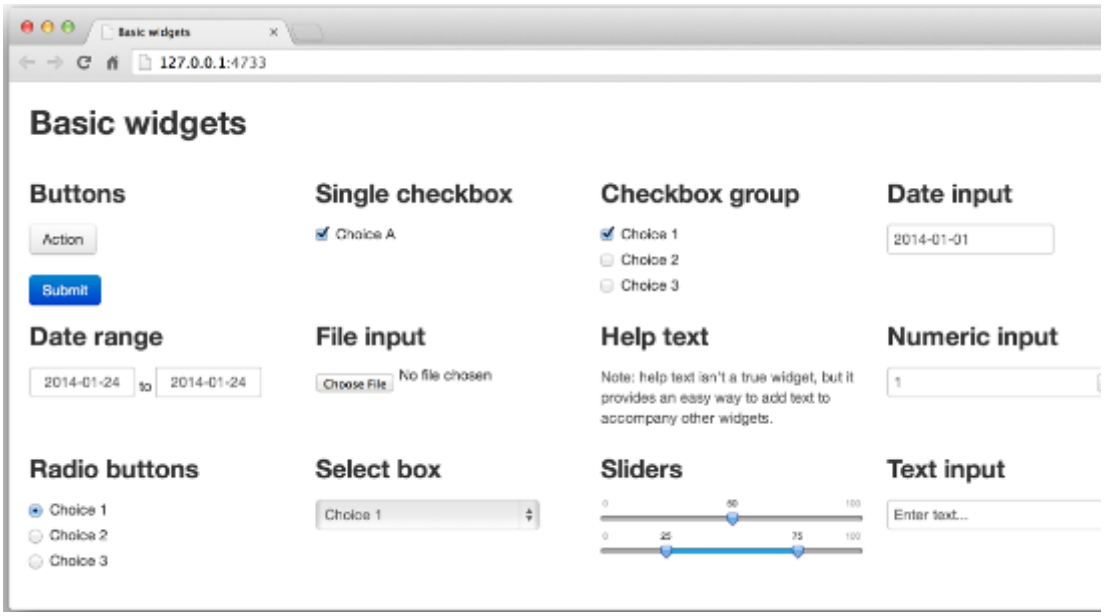

Figura 3.1: Algunos Widgets Básicos

Estas aplicaciones pueden estar alojadas en un servidor externo o en el propio equipo del usuario.

Todo esto, convierte a **shiny** en una poderosa herramienta para la construcción de cuadros de mando donde mostrar los resultados de nuestros desarrollos usando las capacidades analíticas de R, de forma que el receptor de nuestra información pueda interactuar con ella de manera correcta, dinámica y fácil.

3.4.1. Motivación del producto

Para llegar a entender su alcance, es importante conocer el porqué de su desarrollo. Vamos a presentar algunos aspectos sobre que podemos obtener *"de más"* con su utilización.

- En primer lugar, **R**, de por si, ofrece una gran cantidad de excelentes herramientas para manipular, tratar, analizar y representar datos. Sin embargo, **R** se utiliza, esencialmente, de manera local como una aplicación de escritorio.
- Llegados a ese punto, nos planteamos cómo podemos compartir los desarrollos que realicemos en **R** de forma flexible y rápida con otros usuarios.
- Una solución es trasladar **R** al navegador. De esta forma podríamos interactuar dinámicamente con la información, ofreciendose, además, los paquetes de manera online y con uso más fácil para del usuario.

Visualicemos algún ejemplo para ilustrar la idea que estamos introduciendo. Tengamos una *app shiny* que presenta un modelo para estimar la masa en función del volumen y de la edad en masas clonales de Eucalyptus globulus Labill, en la provincia de Huelva.

A través de este cuadro de mandos (figura 3.2) el usuario introduce los datos deseados interactuando con la aplicación.

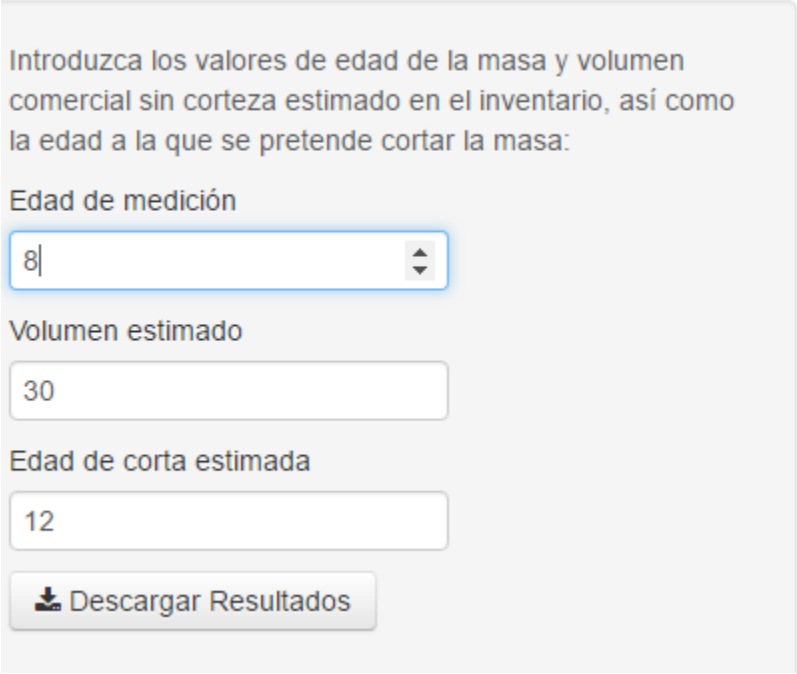

Figura 3.2: Intrucción de valores en un cuadro de mandos. Creado por Julio M. Sevilla Sánchez

El resultado que obtenemos con esos datos determinados es el siguiente:

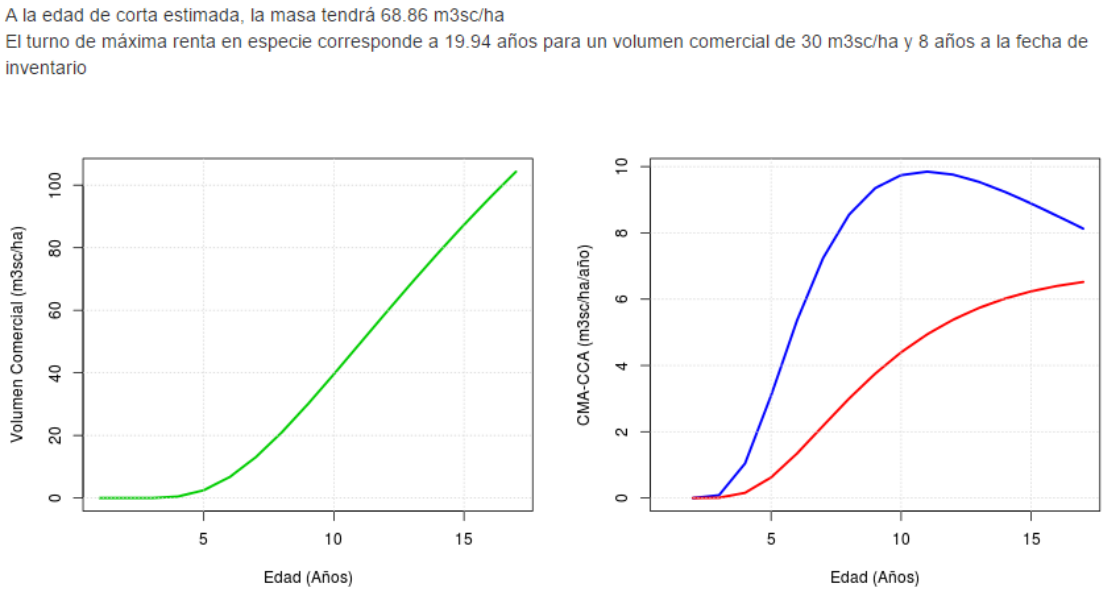

Figura 3.3: Resultado gráfico y texto según los valores introducidos.Creado por Julio M. Sevilla Sánchez

En la figura 3.3, vemos como el resultado es dinámico y se modifica, tanto en texto como en gráficos, según los valores especificados.

3.4.2. Estructura Básica

Una *app* producida a través de **Shiny** se puede dividir en dos componentes:

- **server.R**: Este componente hace referencia a las instrucciones que constituyen los objetos de R de la *app*. Es decir, es el código que se necesita para construir la aplicación.
- **ui.R**: Este componente se encarga de la descripción de la interfaz de usuario de la *app*. Contiene la secuencia de comandos que determina su aspecto y diseño. Gestiona la salida visual y el aspecto de la aplicación.

De esta forma, una aplicacion **Shiny** es un directorio que contiene ambos componentes junto con otros documentos adicionales necesarios (conjuntos de datos, etc,. . .)

3.4.3. Componentes Básicos

Una vez vista su estructura presentamos a tres componentes básicos que permiten construir la **app Shiny**.

La idea básica de **Shiny** es que los valores de entrada por parte del usuario (**inputs**) se conviertan en los resultados o visualizaciones (**outputs**). Para ello el procedimiento de los componentes es el siguiente:

1. Guardar los resultados que se mostrarán al usuario en objetos **output**.
2. Construir los objetos que se deseen mostrar a través de la función **render()** (conexión entre output y input).
3. Acceder a los valores de entrada a través de objetos **input**

Es al fichero **Server.R** es donde se especifica como los valores de entrada (**input**) se convierten en resultados (**output**).

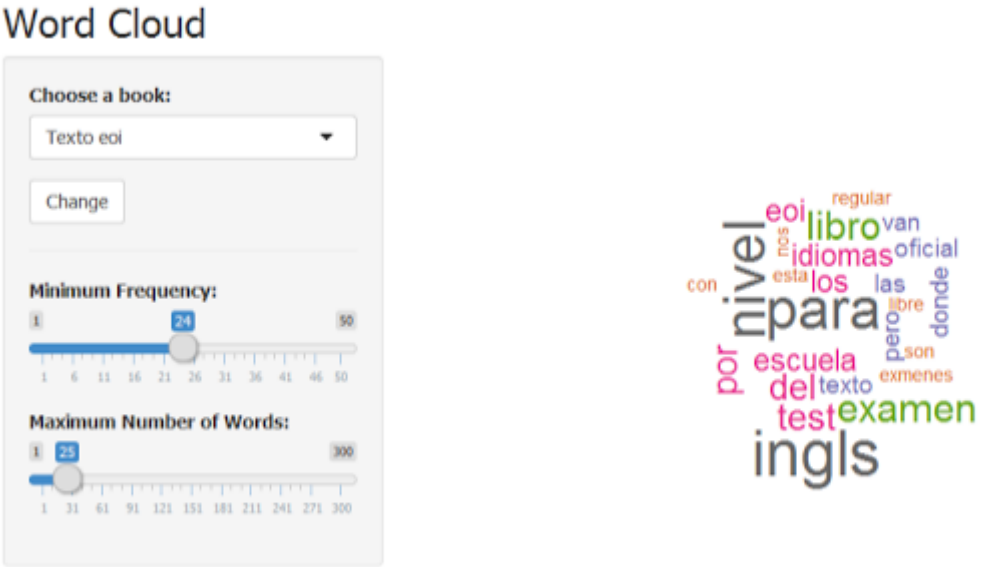

Figura 3.4: Ejemplo aplicación Shiny.

En la figura 3.4 podemos ver un ejemplo de aplicación Shiny donde, seleccionando un libro en concreto, muestra las palabras más repetidas. Permite selecionar por el usuario la frecuencia mínima con la que la palabra debe repetirse para que aparezca, y el número máximo de palabras que queremos que aparezcan.

3.4.4. Compartir la Aplicación

Shiny, junto con **RStudio**, permite presentar la aplicación creada como una página web, de manera que sea accesible en línea.

Para ello dispone de tres opciones:

- **ShinyApss.io**: Permite alojar las *apss* creadas en el servidor de RStudio. Presenta opciones gratuitas y de pago.
- **Shiny Server**: Construye tu propio servidor *Linux* para alojar *apps*. Opción gratis y de código abierto.
- **Shiny Server Pro**: Construye un servidor comercial con autentificación, gestión de recursos y más.

Es decir que, por un lado existe la posibilidad de instalar y configurar nuestro propio servidor Shiny, donde podremos optar por la versión profesional o por una versión gratuita (más limitada) y por otro, también disponemos de una plataforma de hosting en la nube, **ShinyApps.io**. Esta última tiene diferentes modalidades de uso (incluyendo también una versión gratuita) donde podremos poner en marcha nuestra aplicación web programada en R.

3.4.5. Ejemplos Shiny aplicado para la Inteligencia de Negocios

EJEMPLO 1: INDICADORES DEL BANCO MUNDIAL

Con esta aplicación, podemos tener una visualización inteligente sobre una buena parte de los indicadores del Banco Mundial, que son proporcionados por la propia institución. (*figura 3.5*). Las opciones con las que podemos interactuar son:

- Los indicadores están divididos por secciones, mientras que la visualización en el mapa puede ser o *"whole map"* (el mundo entero) o *"By country"* (por país).
- Podemos determinar los indicadores, el año y visualizar los mapas interactivos de diferentes formas, permitiendo la comparación entre países.
- Los gráficos nos permiten interactuar con ellos: hacer zoom y descargar el gráfico como imagen.
- Además, para cada indicador, hay una definición del mismo con su respectiva fuente.

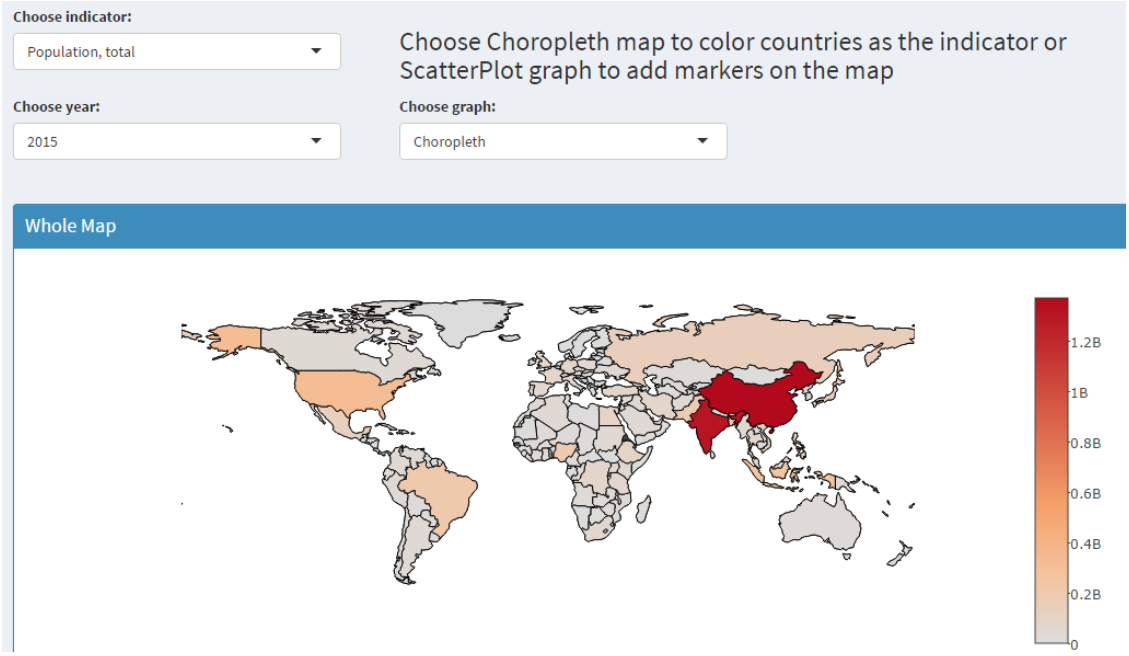

Figura 3.5: Representación de la distribución de la población total en 2015 en el mundo entero mediante intensidad de colores

EJEMPLO 2: CÁLCULOS FINANCIEROS

Calcula valores presentes y futuros, el valor neto de un proyecto y la amortización de un préstamo, según las tasas de interés y los años de devolución que el usuario determine (figura 3.6).

Figura 3.6: Para un valor de 1000 y una tasa de interés del 5 por ciento, si queremos alcanzar un valor de 2000 necesitamos 14.21 años.

De esta forma podemos tener una visualización rápida y clara de las condiciones del mismo, lo que nos ayudará a decidir sobre lo que es más beneficioso para nosotros.

EJEMPLO 3: CUADRO DE MANDOS ECONÓMICO

Presenta un análisis y predicción de datos macroeconómicos, incluyendo comentarios, en diferentes sectores de la economía estadounidense que permite tener una idea rápida de la actividad económica, donde el usuario puede, por ejemplo, acotar las fechas haciendo *zoom* en el periodo que resulte de interés.

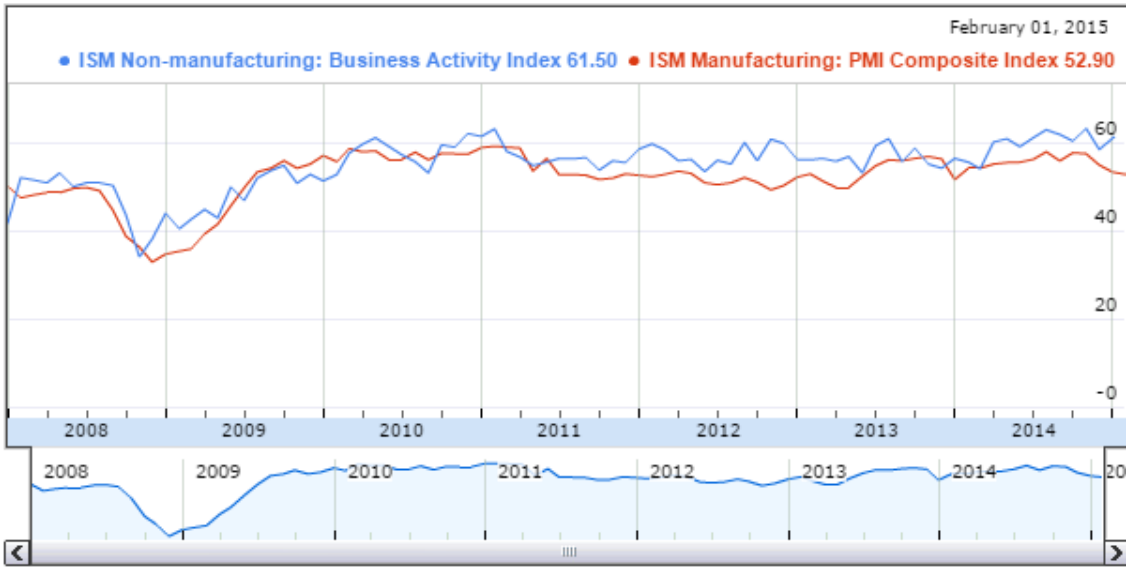

Figura 3.7: Evolución de los valores de los índices de actividad económica para actividades manufacturadas y no manufacturdas

3.4.6. Aclaración de Conceptos Mencionados.

3.4.6.1. Programación Reactiva

El concepto de **programación reactiva** se puede definir como :

> "Es un paradigma de programación que está orientado a los flujos de datos y más específicamente a la propagación de los cambios de los datos en dicho flujo"

La **programación reactiva** es un modelo de programación para aplicaciones robustas y muy estables en Internet. Está orientada especialmente a la programación de los cambios de los datos en los flujos.

Esta idea la podemos visualizar de manera más clara con un ejemplo.

En primer lugar veamos un ejemplo de *programación no reactiva*

```
var a = 1;
var b = 2;
var c = a + b;

console.log(c); // 3

a = 2;

console.log(c); // 3
```

Figura 3.8: Ejemplo programación no reactiva

Vemos, en la figura 3.8, que una vez guardado el valor de c, si cambiamos el valor de a posteriormente, no se modifica.

```
var a = 1;
var b = 2;
var c = a + b;

console.log(c); // 3

a = 2;

console.log(c); // 4
```

Figura 3.9: Ejemplo programación reactiva

Es decir, que la **programación reactiva** permite obtener resultados dinámicos, como vemos en la figura 3.9, donde al cambiar el valor de a, este cambio se propaga a c ya que depende de a.

De esta forma, centrándonos en su papel en la **app shiny**, la **Reactividad** permite reconstruir la salida (*output*) si se cambia una entrada(*input*) de la que dependa.

3.4.6.2. Widget

Los **widgets** son pequeñas aplicaciones o programas que facilitan el acceso a funciones usadas frecuentemente por los usuarios y que proporcionan información de forma visual, rápida e intuitiva.

Centrándonos en su uso en **Shiny**, a través de estos elementos web, los usuarios pueden interactuar con el **widget**, enviando, así, mensajes a la **app shiny**. De esta forma, cuando el usuario cambie el **widget**, el valor cambiará también.

3.4.6.3. RMarkdown

Markdown es una forma sencilla de agregar formato a textos web, como itálica, negrita, listas, etc. Se podría decir que es una forma de crear páginas web sin tener experiencia en HTML y que permite aprovechar las ventajas del texto plano sin tener que renunciar al formato.

A través del paquete **RMarkdown** [6] podemos escribir informes reproducibles y dinámicos con **R**, con la opción de incluir código R dentro de ellos y presentar resultados en presentaciones, documentos, PDF, HTML, Word y más.

3.5. Librería HTMLWidgets.

El paquete **htmlwidgets** proporciona widgets en HTML que funcionan de forma similar a como lo hace un *plot* en **R**, pero proporcionando una gran ventaja: producen visualizaciones web interactivas.

Es muy sencillo de utilizar, de forma que con una o dos líneas de código en **R** podemos producir visualizaciones impresionantes, como gráficos en 3D o en mapas.

Este paquete permite utilizar los widgets tanto en la propia consola de R, así como dentro de aplicaciones web **shiny** o en informes **RMarkdown**.

Es una manera rápida y fácil de crear gráficos interactivos basados en el lenguaje javascript sin saber nada de este.

3.5.1. Widgets destacados

htmlwidgets dispone de una gran cantidad de librerías que permiten crear atractivas visualizaciones de los datos de manera interactiva en la web. Como cabría esperar, esta interfaz requiere de la utilización del lenguaje *JavaScript* para su creación, lo cual haría que fuera de difícil acceso y utilización para la mayoría de profesionales estadísticos y analistas que no dispongan de este conocimiento. Sin embargo, **htmlwidgets** provee múltiples librerías que permiten a cualquier usuario crear una interfaz de forma muy clara y directa.

La compresión de **htmlwidgets**, y saber cómo aprovechar las ventajas que nos presentan los paquetes como *Leaflet* o *DT* por ejemplo, nos ayudará a crear visualizaciones impresionantes, interactivas y convincentes, pero sobre todo, con la ventaja de la utilización de muy poco código.

A continuación vamos a presentar por las principales y más destacadas librerías que permiten al usuario explorar los datos y la información de manera dinámica, así como combinarla con la visualización de gráficos de interactividad directa .

3.5.1.1. Librería Plotly: Plataforma de gráficos interactivos.

La librería **Plotly** [7] crea gráficos interactivos, permitiendo una publicación online de calidad.

Permite generar una gran cantidad de gráficos: plots, scatter plots, area charts, bar charts, error bars, box plots, histograms, heatmaps, subplots, multiple-axes, polar charts and bubble charts.

[6] http://rmarkdown.rstudio.com/
[7] https://plot.ly/r/

CARACTERÍSTICAS

- **Son Interactivos**. Permiten hacer *zoom*, cambiar la escala, y movernos por el gráfico.

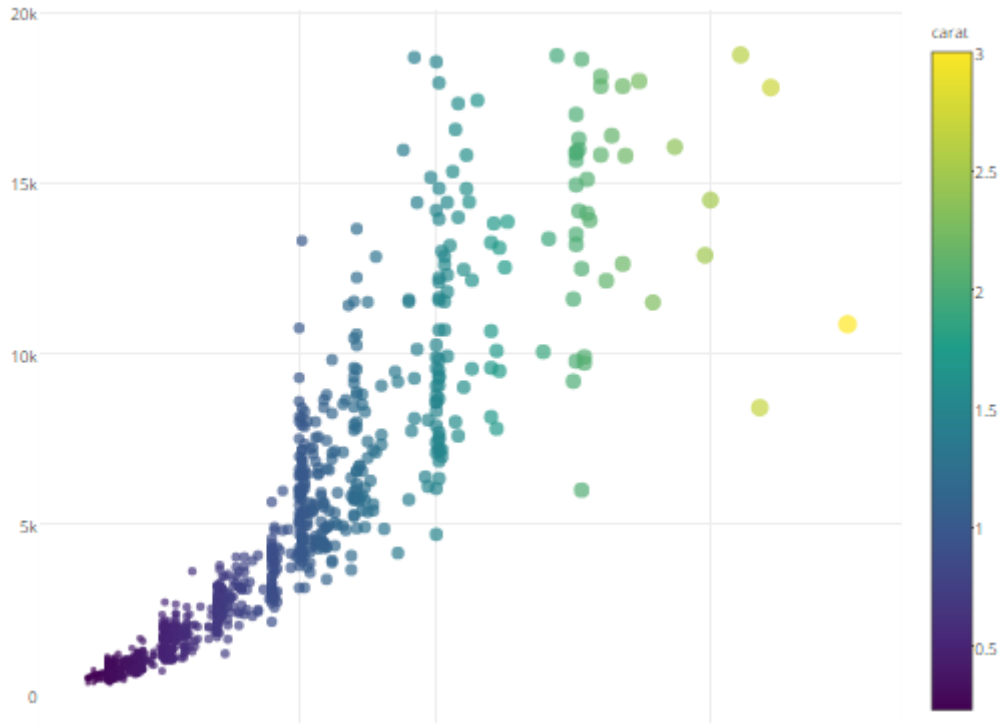

Figura 3.10: Ejemplo gráfico utilizando Plotly

- **Integración con ggplot2**. Con la función *ggploty* podemos convertir un objeto ggplot2 a un objeto ploty.

```
p <- ggplot(data = d, aes(x = carat, y = price)) +
  geom_point(aes(text = paste("Clarity:", clarity))) +
  geom_smooth(aes(colour = cut, fill = cut)) + facet_wrap(~ cut)

ggplotly(p)
```

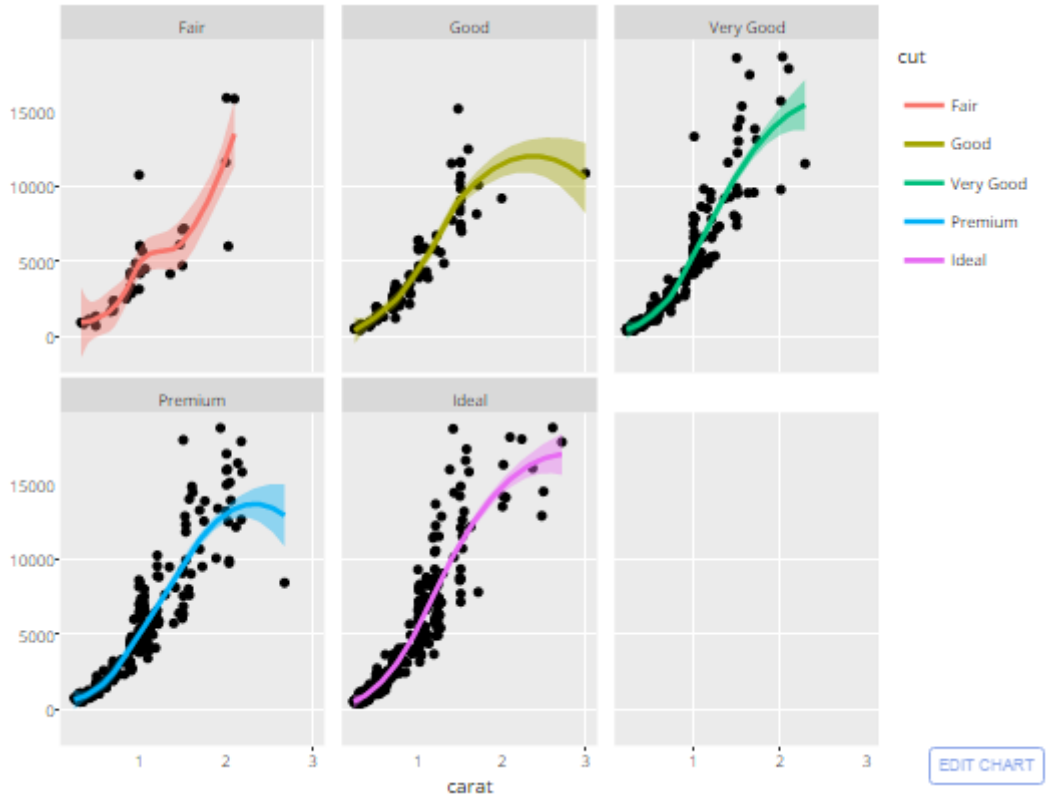

Figura 3.11: Creación de objeto plotly a través de gráfico construido con ggplot2

- **Opciones de salida y estilo**. Presentan una gran flexibilidad a la hora de su representación. Permite modificar e incorporar elementos a nuestro gráfico: Estilo de la fuente, ejes en 3D, textos y anotaciones, tamaño del gráfico, leyenda, etiquetas de los ejes, incorporación de Latex en los gráficos, etc.
- **Plotly y Shiny**. Desde que **Plotly** está adherido al marco de trabajo **htmlwidgets** es muy fácil incorporar sus gráficos en aplicaciones **shiny**. Esto facilita que los gráficos *plotly* cambien de forma dinámica según los valores de entrada dados por el usuario.

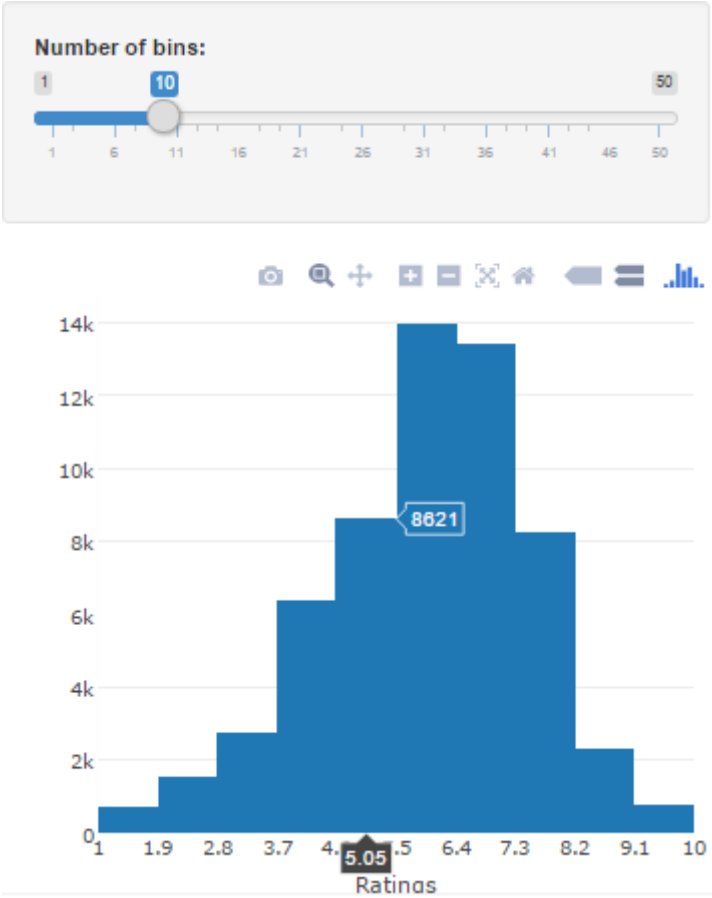

Figura 3.12: Plotly + Shiny

EJEMPLO DE GRÁFICOS CON PLOTLY

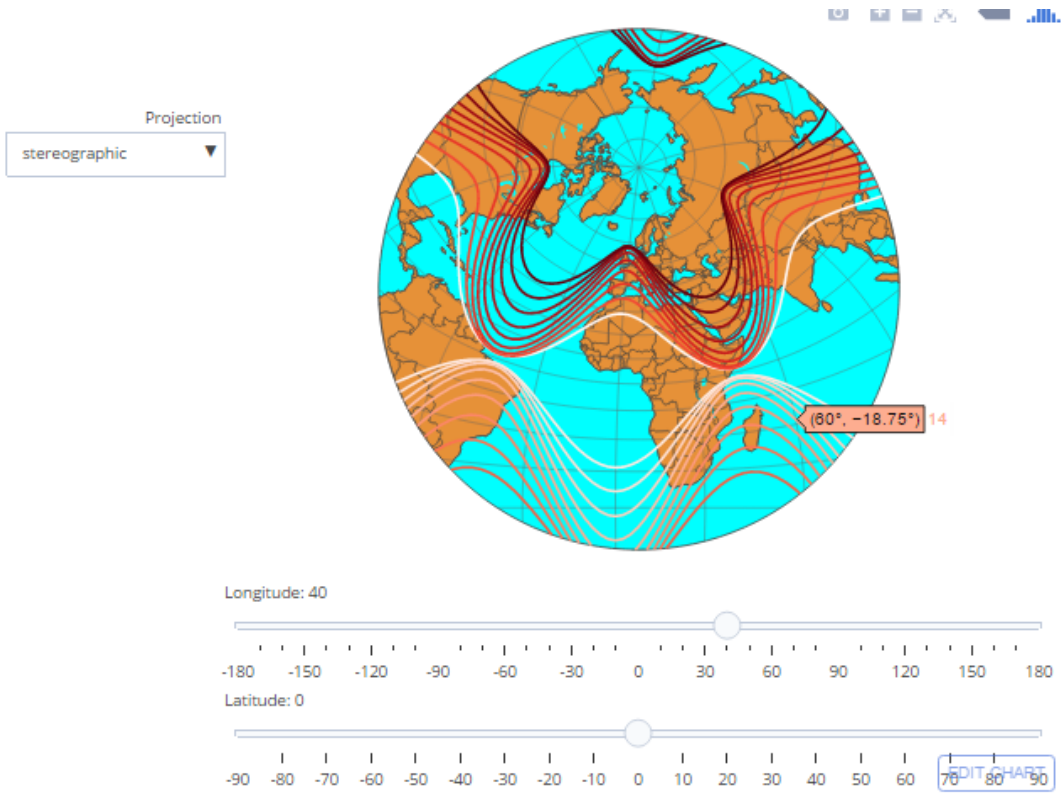

Figura 3.13: Gráfico con controles de desplazamiento de barras múltiples.

Figura 3.14: Mixtura de subgráficos: Abertura de nuevas tiendas de la cadena Walmart por año

3.5.1.2. Librería Leaflet: Introducir mapas en R.

Leaflet [8] es una librería de JavaScript muy popular para la creación de mapas interactivos. Mediante el paquete, desarrollado para **R**, podemos controlar e integrar mapas de Leaflet en R, importando los mapas que deseamos visualizar en nuestro mapa web.

Permite crear mapas dinámicos, como vemos en figura 3.15, que soportan zoom y movimientos sobre él (*panning*), y añadir anotaciones del tipo: marcador, figuras y ventanas emergentes.

[8] https://rstudio.github.io/leaflet/

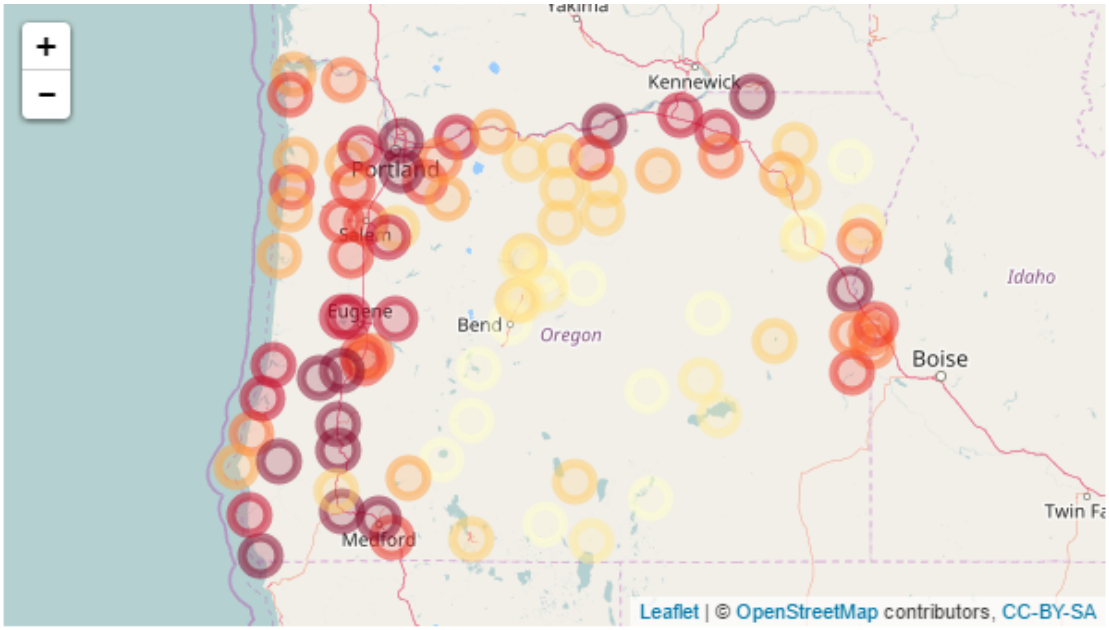

Figura 3.15: Gráfico con Leaflet

CARACTERÍSTICAS

- Incorpora mapas en **Rmarkdown**.
- Fácil representación de objetos espaciales desde el paquete **sp** o a partir de conjuntos de datos (*data frames*) con columnas de información sobre latitud/longitud.
- Es posible visualizar más de un mapa a la vez mediante capas (figura 3.16).

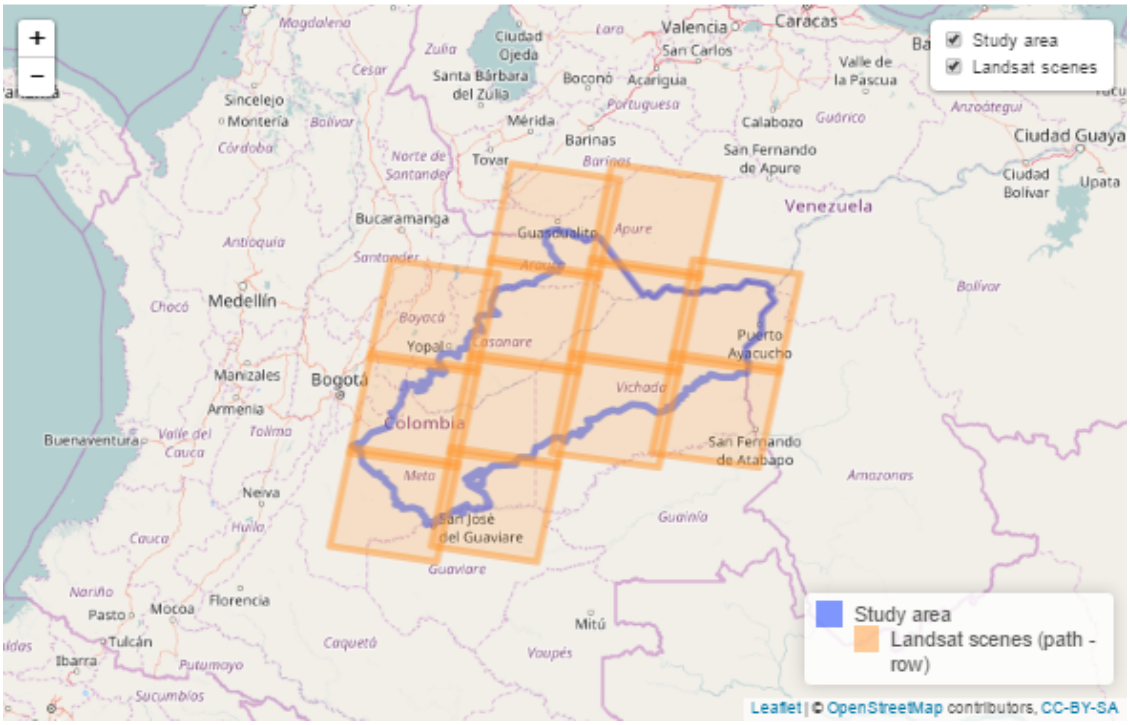

Figura 3.16: Gráfico con diferentes capas simultáneas

- Se pueden ir añadiendo elementos al mapa mediante capas (*layers*) a través de funciones como *addMarkers*, *addTiles()*, etc. De esta forma modificamos el widget.

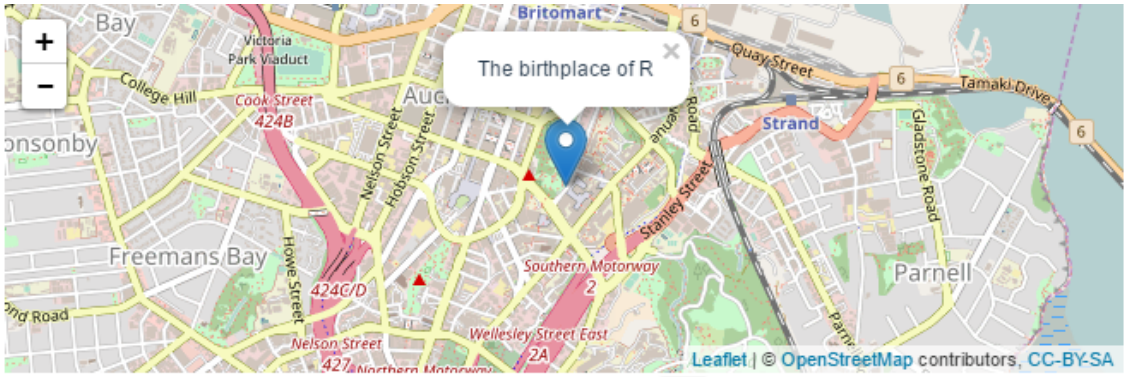

Figura 3.17: Gráfico con marcador señalando el lugar de nacimiento de R. Se incorpora mediante coordenadas

- Integración con **Shiny** (figura 3.18).

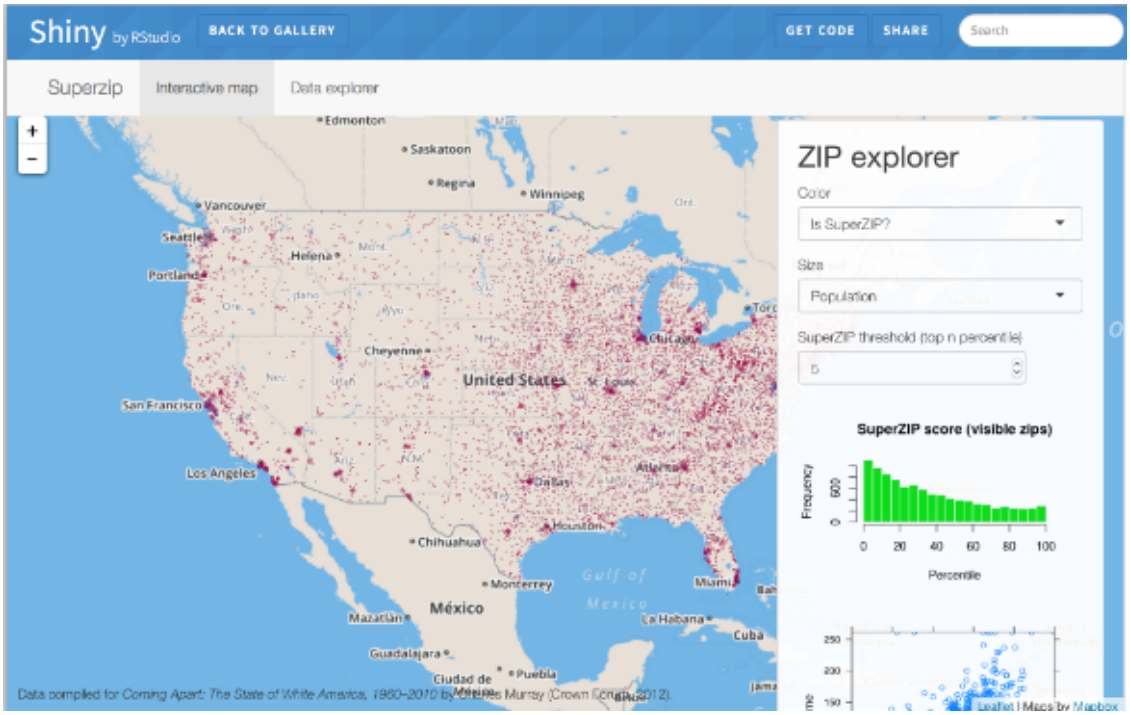

Figura 3.18: Leaflet + Shiny

3.5.1.3. Librería DT: Tabla de datos interactivas.

La librería **DT** [9] permite disponer matrices o data frames de código **R** como tablas HTML de forma interactiva.

Provee una interfaz en **R** para la librería de *JavaScript* conocida como **DataTables**.

CARACTERÍSTICAS:

- Permite realizar múltiples tratamientos sobre los datos: filtrar, muestra de filas, búsquedas, ordenación de las columnas, alteración de posición de las columnas, mostrar por páginas etc.

[9] https://rstudio.github.io/DT/

Show 10 entries Search:

	Sepal.Length	Sepal.Width	Petal.Length	Petal.Width	Species
1	5.1	3.5	1.4	0.2	setosa
2	4.9	3	1.4	0.2	setosa
3	4.7	3.2	1.3	0.2	setosa
4	4.6	3.1	1.5	0.2	setosa
5	5	3.6	1.4	0.2	setosa
6	5.4	3.9	1.7	0.4	setosa
7	4.6	3.4	1.4	0.3	setosa
8	5	3.4	1.5	0.2	setosa
9	4.4	2.9	1.4	0.2	setosa
10	4.9	3.1	1.5	0.1	setosa

Showing 1 to 10 of 150 entries Previous 1 2 3 4 5 ... 15 Next

Figura 3.19: Tabla creada con DT

- La función principal en este paquete es **datatable()**. Esta función permite crear un **widget** HTML para mostrar objetos de datos de R.
- Al ser una forma de mostrar un objeto de R como *HTML*, tiene muchas posibilidades y opciones que permiten personalizar las tablas.
  - Opciones de diseño a partir de argumentos CSS: bordes de las columnas y filas. Esta opción está asociada al argumento *class* (figura 3.20). Permite usar combinaciones de las opciones para construir el estilo de tabla que queremos.Alguno de los valores que puede tomar son:

cell-border: Borde alrededor de los cuatros lados de cada celda.

compact: Reduce la cantidad de espacio en blanco, aumentando la densidad de información en la pantalla.

nowrap: Todo el texto de las celdas está en una sola línea.

Show 10 entries Search:

	Sepal.Length	Sepal.Width	Petal.Length	Petal.Width	Species
1	5.1	3.5	1.4	0.2	setosa
2	4.9	3	1.4	0.2	setosa
3	4.7	3.2	1.3	0.2	setosa
4	4.6	3.1	1.5	0.2	setosa
5	5	3.6	1.4	0.2	setosa
6	5.4	3.9	1.7	0.4	setosa

Showing 1 to 6 of 6 entries

Figura 3.20: class = cell-border stripe

- Manejo de las filas: No mostrar nombres, tomar la primera columna como nombre, o cambiarlo a nuestro gusto.
- Manejo de columnas: Cambiar nombres de todas o algunas, como podemos ver en la figura 3.21.

Show 10 entries Search:

	Here	Are	Some	New	Names
1	5.1	3.5	1.4	0.2	setosa
2	4.9	3	1.4	0.2	setosa

Figura 3.21: Ejemplo con nombre de columnas diferentes

- Y muchas otras opciones como añadir título en la posición que queramos, cambiar formato de la cabecera, modificar el tipo y la visualización del filtrado, cambiar color y relleno de las celdas(3.22), formato de las columnas (3.23), mostrar botones, scroller, etc..

Show 10 entries Search:

	Sepal.Length	Sepal.Width	Petal.Length	Petal.Width	Species
1	5.1	3.5	1.4	0.2	setosa
2	4.9	3	1.4	0.2	setosa
3	4.7	3.2	1.3	0.2	setosa
4	4.6	3.1	1.5	0.2	setosa

Figura 3.22: Relleno color cabecera

A	B	C	D	E
-€460,476	-€967,824	-594707	0.6478935	99.509
-€130,177	-€117,975	-107917	0.3198206	97.691

Figura 3.23: Columna A y B euros, columna E 3 decimales

Show 10 entries Search:

	Sepal.Length	Sepal.Width	Petal.Length	Petal.Width	Species
1	5.1	3.5	1.4	0.2	setosa
2	4.9	3	1.4	0.2	setosa
3	4.7	3.2	1.3	0.2	setosa
4	4.6	3.1	1.5	0.2	setosa
5	5	3.6	1.4	0.2	setosa
6	5.4	3.9	1.7	0.4	setosa
7	4.6	3.4	1.4	0.3	setosa
8	5	3.4	1.5	0.2	setosa
9	4.4	2.9	1.4	0.2	setosa
10	4.9	3.1	1.5	0.1	setosa

Showing 1 to 10 of 150 entries Previous 1 2 3 4 5 ... 15 Next

Figura 3.24: Rotación de texto, color de números en Sepal.Width según su valor(blanco si valor menor que 3.4, azul si entre 3.4 y 3.8, rojo mayor que 3.8), color de relleno de celda según valor (gris si menor que 3.4, amarillo en caso contrario)

- Integración con **Shiny**, de manera que se dinamiza la información que muestra la tabla según el interés del usuario, por ejemplo el número de los índices que se muestran en la página actual, o como las posibilidades que podemos ver en la 3.25 y 3.26.

Search:

	speed	dist
1	4	2
2	4	10
3	7	4
4	7	22
5	8	16
6	9	10
7	10	18
8	10	26
9	10	34
10	11	17

Previous 1 2 3 4 5 Next

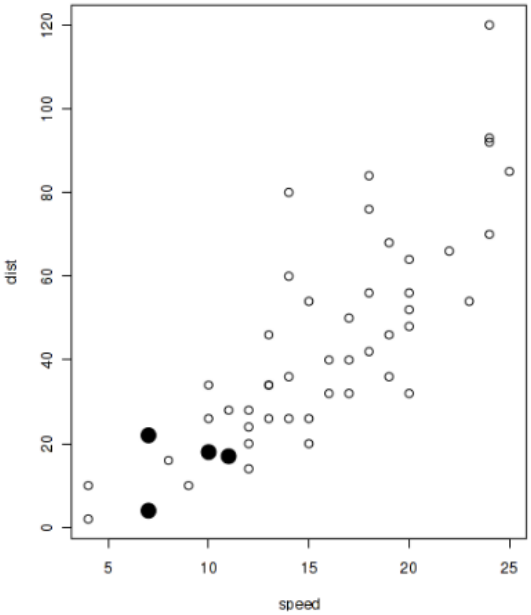

Figura 3.25: Resalta en el gráfico las filas señaladas

Search: m

	hp	mpg
Mazda RX4	110	21
Mazda RX4 Wag	110	21
Merc 240D	62	24.4
Merc 230	95	22.8
Merc 280	123	19.2
Merc 280C	123	17.8
Merc 450SE	180	16.4
Merc 450SL	180	17.3
Merc 450SLC	180	15.2
Chrysler Imperial	230	14.7

Previous 1 2 Next

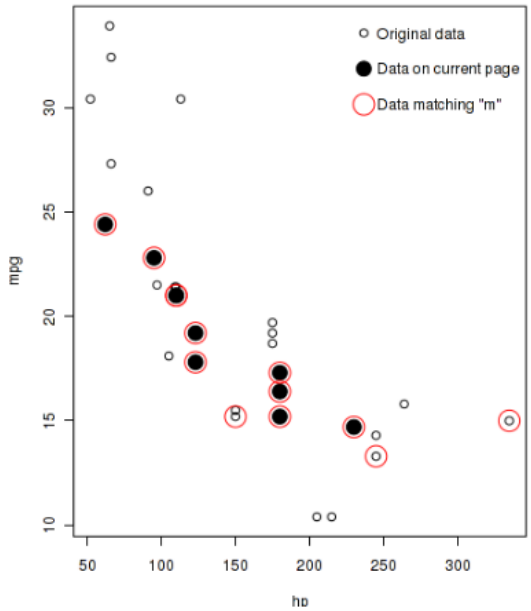

Figura 3.26: Muestra todos los datos, marca los casos de esa página y resalta los casos cuya fila contiene una m

3.5.1.4. Librería dygraphs: Series temporales.

El paquete **dygraphs** [10] es una interfaz de **R** para integrar la librería gráfica de *dygraphs de JavaScript*. Provee al usuario importantes facilidades y posibilidades para la representación de series temporales en R.

[10] https://rstudio.github.io/dygraphs/

Son llamativas e interactivas que se generan de manera muy sencilla con únicamente una o dos líneas de código.

CARACTERÍSTICAS

Incluye muchos aspectos interesantes para mostrar gráficamente objetos de series temporales en **R**

- "Dibuja", de manera automática, objetos **xts**[11], y objetos convertible a la estructura xts, de series temporaless.
- Presenta interesantes características para interactuar con la serie temporal: *zoom*, *resaltar* puntos de la serie al pasar el ratón, etc.

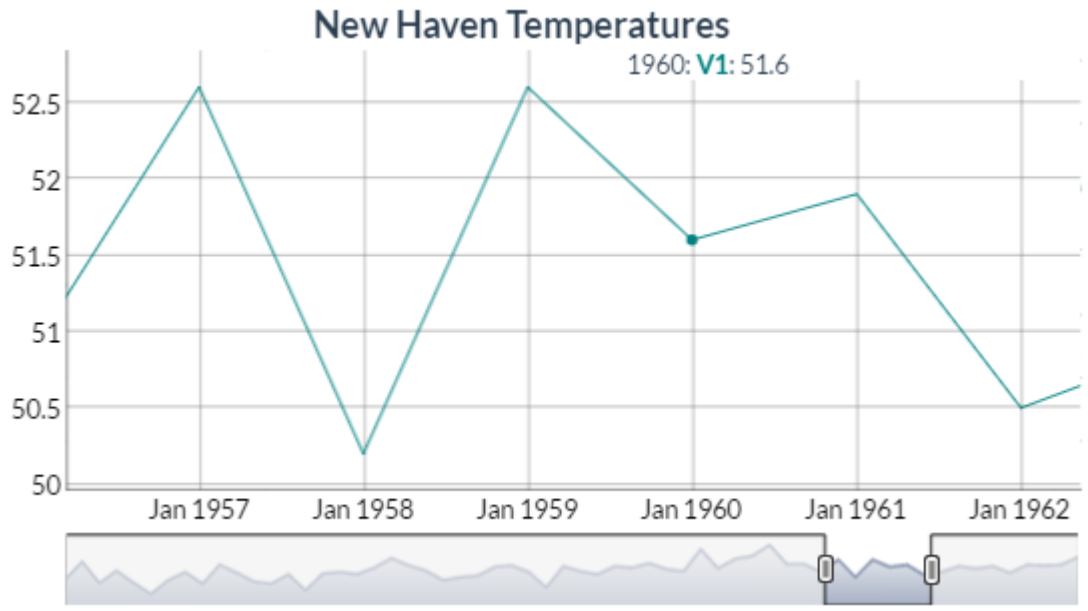

Figura 3.27: Argumento opcional (Range selector) para mostrar un selector de rango de fechas que incluye en la interfaz, con un uso muy claro, la posibilidad de hacer zoom al rango de fecha que deseamos

- Permite personalizar la gráfica a través de un fichero CSS externo (*con argumento dyCSS*)
- Múltiples opciones de configuración respecto a los ejes y la motorización de la serie: *Grosor de los ejes (figura 3.28), etiquetas, relleno bajo la línea de la serie (figura 3.28), rejilla de fondo,color de los ejes, doble escala en eje y (figura 3.29), etc.*

[11] xts es un paquete que permite manipular series temporales en R a través de potentes herramientas que hacen de está tarea un procedimiento más rápido y libre de errores

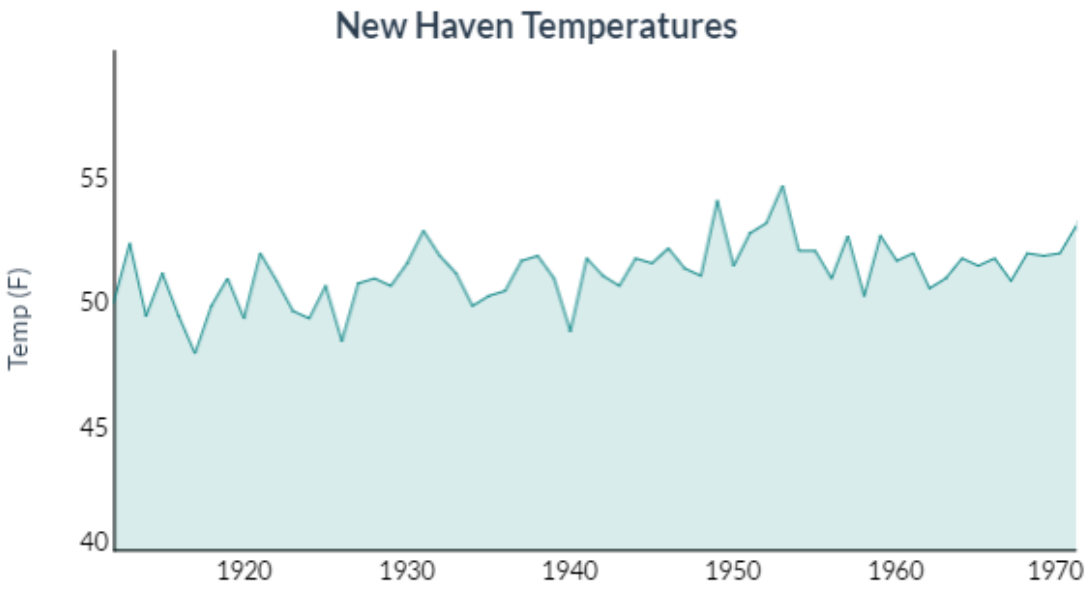

Figura 3.28: Ejes más gruesos y relleno (mediante argumentos dyAxis y dyOptions)

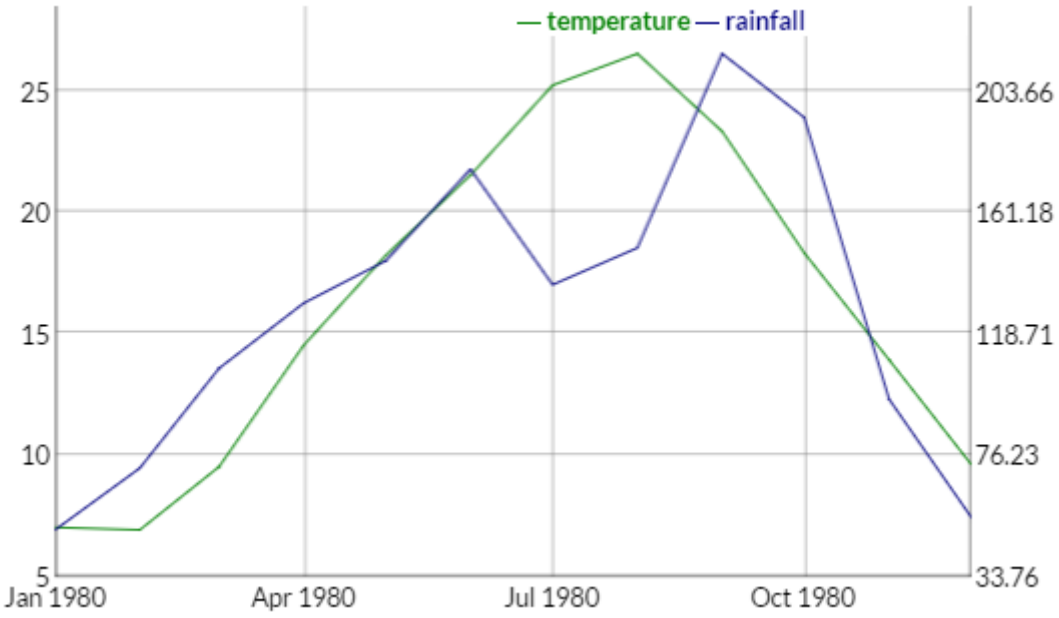

Figura 3.29: Ejemplo de dos series superpuestas con diferente escala (mediante argumentos dyAxis)

- Permite incluir información gráfica sobre los datos y su confianza: *medidas de error y/o intervalos de confianza.(figura 3.30)*

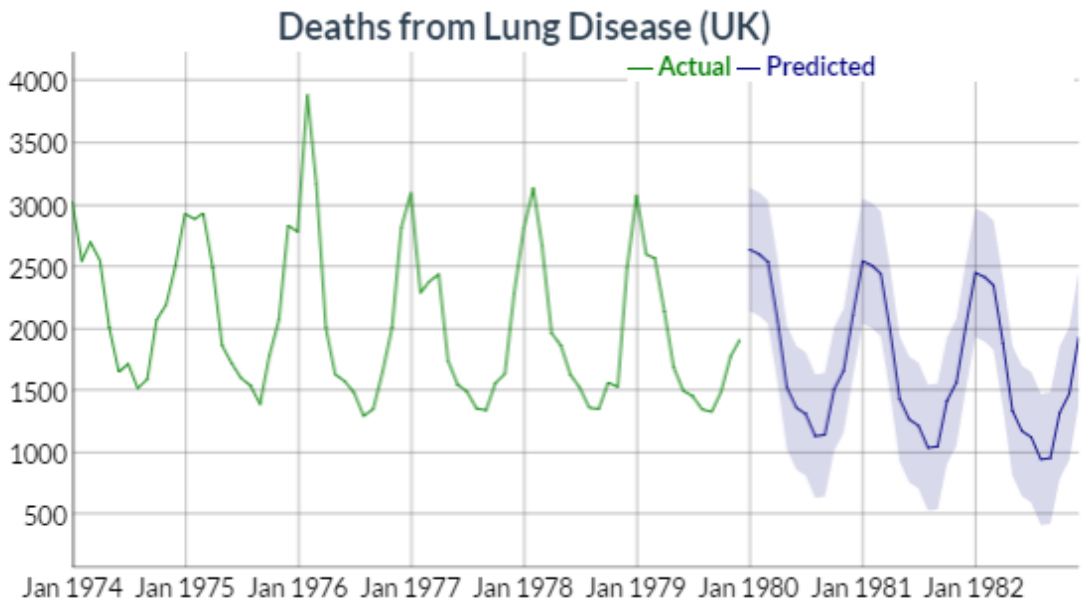

Figura 3.30: Valor real de una serie temporal y, a continuación, su predicho con el correspondiente intervalo de confianza (mediante argumento dySeries)

- Dispone de varias opciones gráficas para superponer sobre la serie: *regiones sombreadas (figura 3.31), marcación de eventos (figura 3.33), anotaciones (figura 3.32), etc*

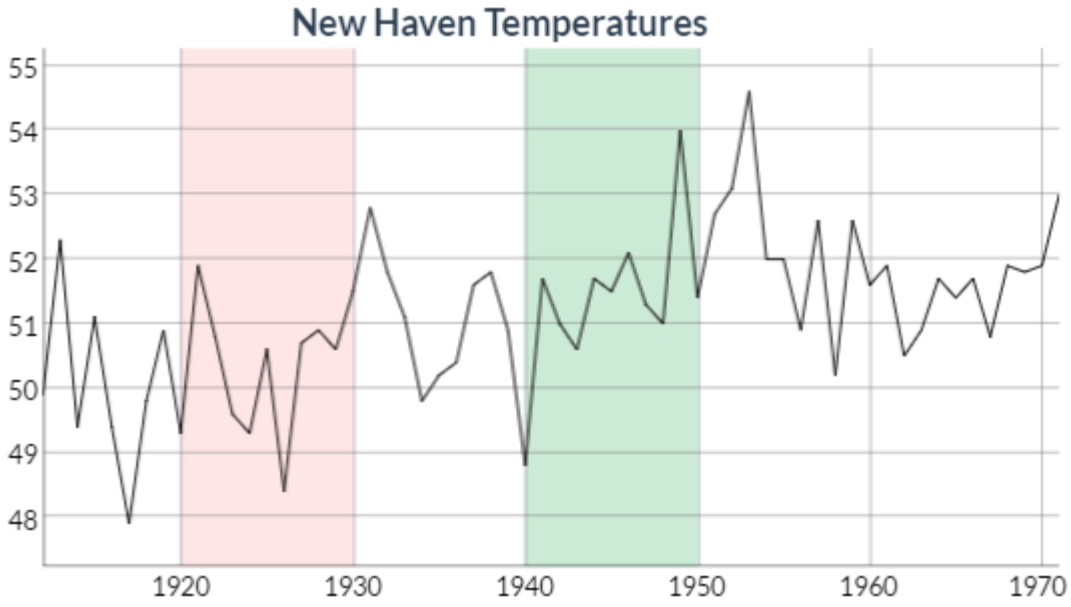

Figura 3.31: Sombreado de zonas. Rosado desde 1920 a 1930, celeste desde 1940 hasta 1950 (mediante argumento dyShading)

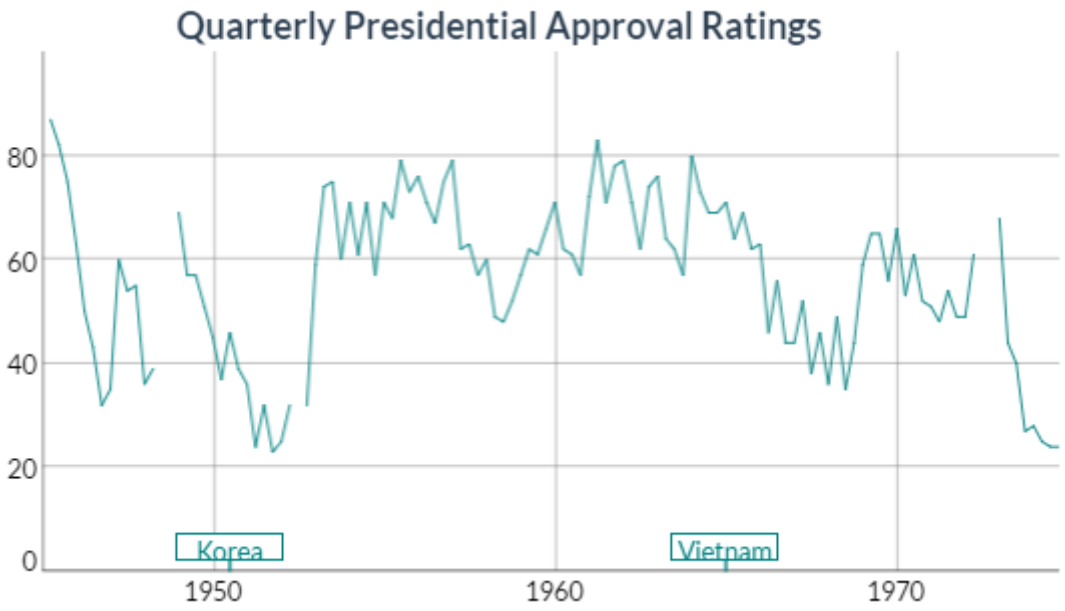

Figura 3.32: Anotaciones de texto mediante argumento presAnnotation

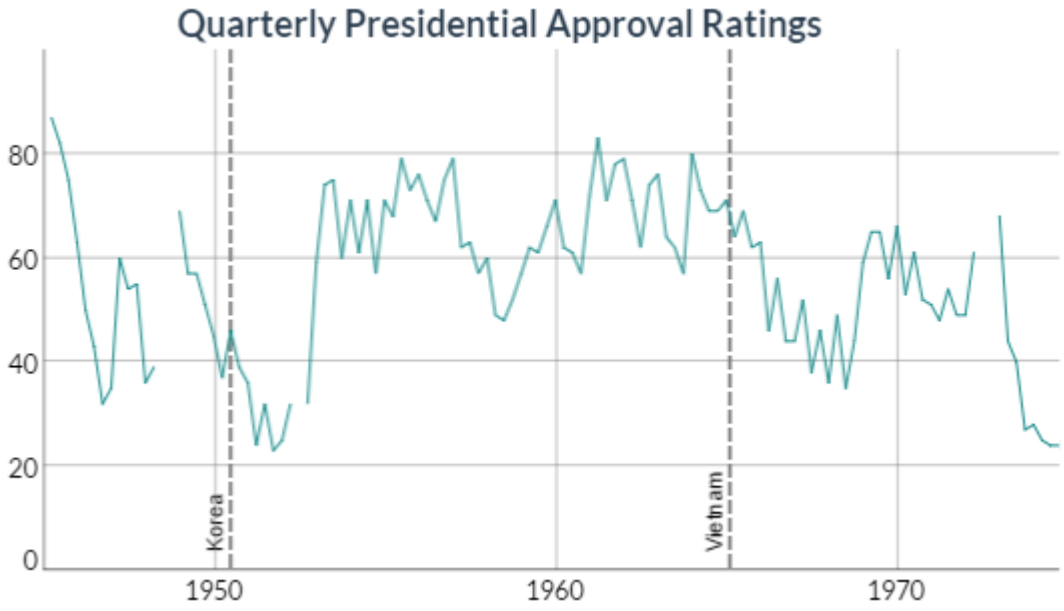

Figura 3.33: Señalación de eventos: Fechas en que se produjo el primer despliegue de tropas estadounidenses tanto en Corea como en Vietnam

- Y otras muchas más opciones que permiten personalizar las gráficas:

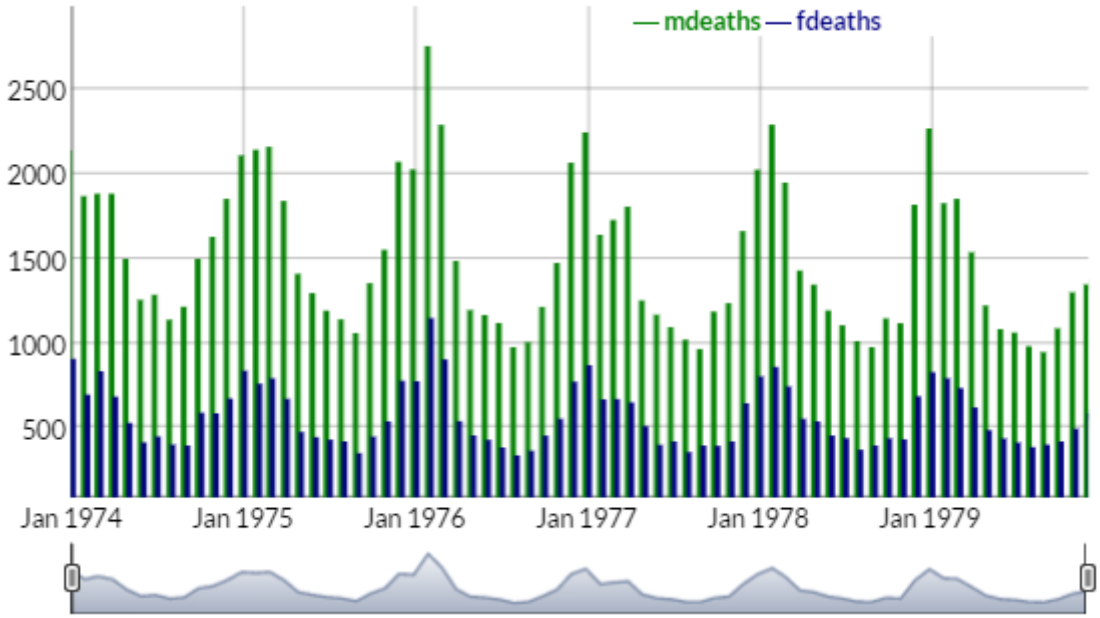

Figura 3.34: Representación de dos series temporales mediante barras en vez de una línea que respresente sus perfiles

Figura 3.35: Sobre un punto se da diferente información, valor más bajo, más alto, media.

- Posibilidad de incorporarlo a documentos **RMarkdown** y aplicaciones web de **Shiny** (figura 3.36).

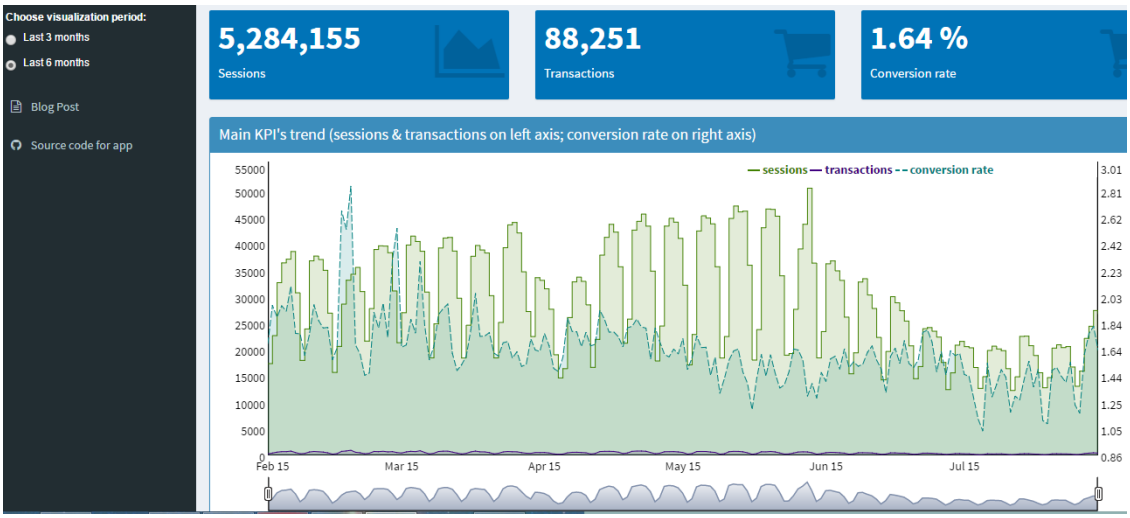

Figura 3.36: Shiny y dygraph: Permite seleccionar el periodo de visualización de los meses.

3.5.1.5. Librería D3HEATMAP: Mapas de calor interactivos.

El paquete **d3heatmap** [12] permite generar *mapas de calor* utilizando el soporte *D3*, permitiendo opción de *zoom* y *resaltado* de filas o columnas.

CARACTERÍSTICAS

- Resaltas filas o columnas *clicando* sobre los ejes
- "Arrastrar y mover" sobre el gráfico para hacer zoom.
- Opciones para hacer *clustering* y *dendogramas* (figura 3.37)

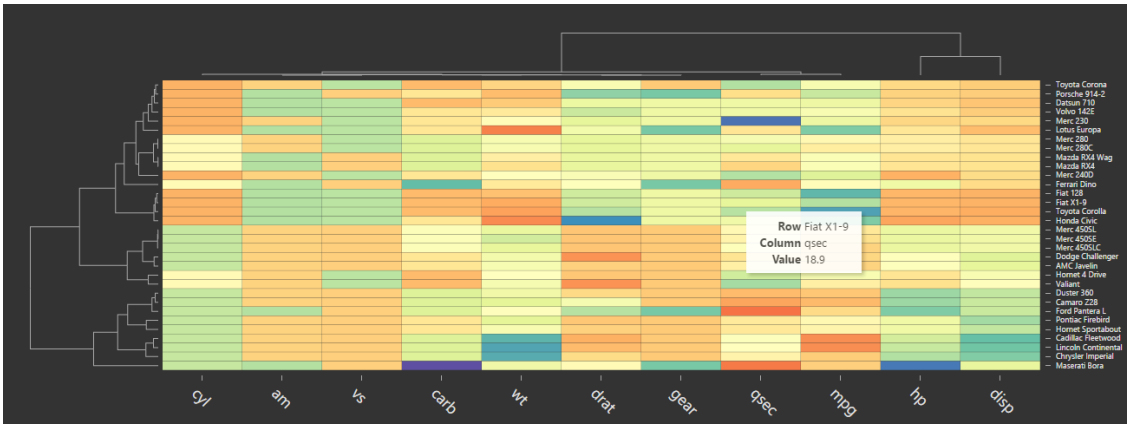

Figura 3.37: Heatmap con dendograma

3.5.1.6. Librería THREEJS: Gráficos 3D interactivos.

El paquete **threejs** [13] incluye representaciones gráficas en 3D: *scatterplot* y del globo terráqueo, usando la estructura *JavaScript* de *three.js* y el paquete **htmlwidgets** de **R**.

[12] https://github.com/rstudio/d3heatmap
[13] https://github.com/mrdoob/three.js/

CARACTERÍSTICAS

- Widget para **R** y **Shiny**.
- Permite manipular los gráficos, por ejemplo cambiar la perspectiva de su visión, como podemos ver en la figura 3.38, simplemente con el ratón.

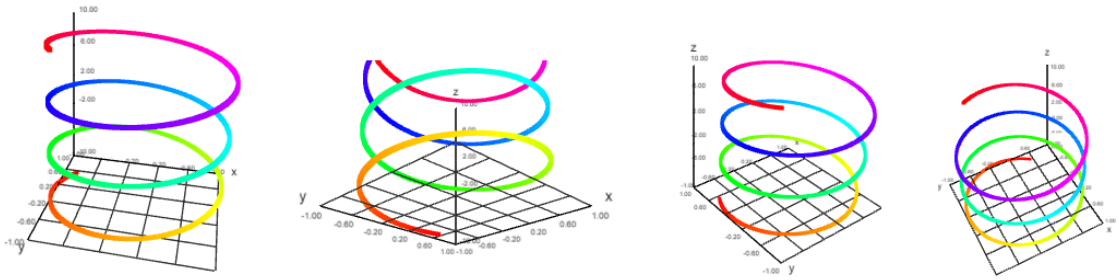

Figura 3.38: Diferentes perspectivas de un gráfico

- Los widgets que se ofrecen son fácil de usar y utilizar directamente en **RStudio**, **RMarkdown** y **Shiny**.

Vamos a ver dos ejemplos para mostrar las posibilidades tan vistosas de las cuales dispone este paquete.

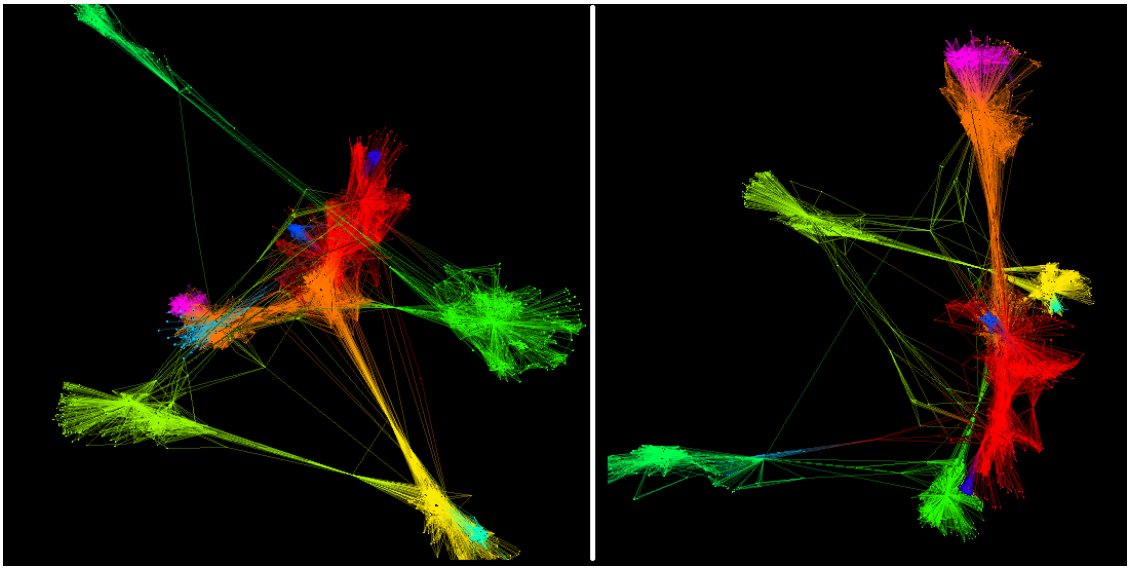

Figura 3.39: Ejemplo de red

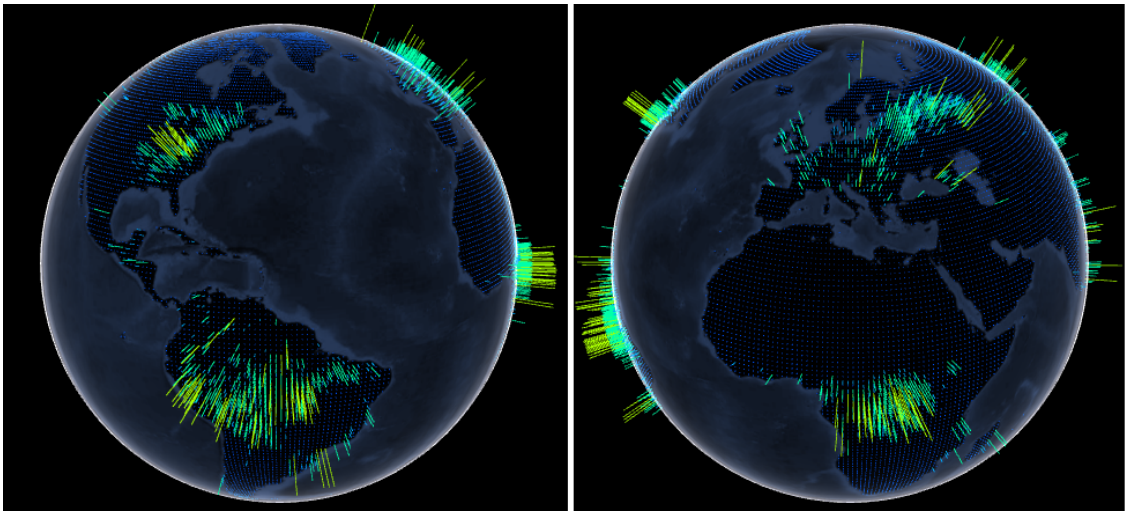

Figura 3.40: Representación del índice de vegetación global(NASA MODIS)

3.6. Librería Flexdashboard

Flexdashboard es un paquete de **R**, bastante reciente, creado por **RStudio**. Con este paquete vamos a ser capaces de crear fácilmente *cuadros de mando* flexibles, atractivos e interactivos con **R**.

La creación y personalización de los *cuadros de mando* es realizada a través de **RMarkdown**, pudiendo, opcionalmente, incluir componentes de **Shiny** para añadir una interactividad extra (3.41).

Es una alternativa más simple a la utilización de **Shiny** que puede llegar a ser demasiado complejo, aunque presenta el incoveniente de que en ocasiones puede ser menos robusta.

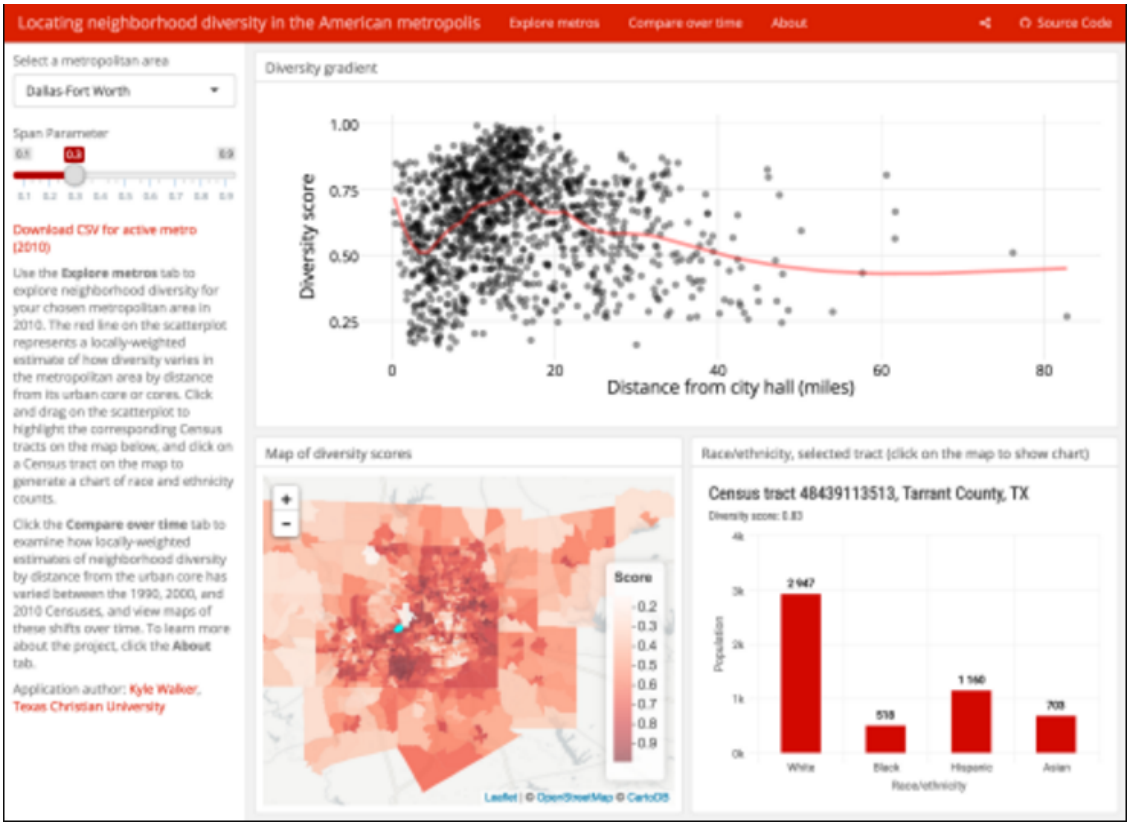

Figura 3.41: Flexdashboard y Shiny

Por lo tanto, podemos usar **flexdashboard** para publicar visualizaciones relacionadas a un conjunto de datos mediante *cuadros de mando*.

3.6.1. Características principales.

Lo más destacado del paquete **flexdashboard** incluye lo siguiente:

- Soporte para una amplia cantidad de componentes: **widgets** interactivos a través del paquete **htmlwidgets**, tablas tabuladas, gráficos, cajas de valor, medidas descriptivas, etc.
  a) Utilización de **htmlwidgets**: Interacción con *JavaScript* para visualizar la información de los datos.
  b) Tablas: Con opciones de filtro, orden y paginado.
  c) Cajas de valor que resumen descriptivamente los datos, mostrando así las medidas más destacadas.
- Presenta una manera muy flexible y fácil para especificar la colocación y estructura del cuadro de mando (filas y columnas). Los componente son *re-escalados*, de forma inteligente y automática, para ajustarse al navegador y adaptarse a la salida de dispositivos móviles.
- Dispone de un amplio soporte para realizar anotaciones sobre el **cuadro de mandos**.
- Estructura **Storyboard**. Esto permite visualizar el cuadro de mando a través de *cajas en secuencia*, como vemos en la figura 3.42, que se van seleccionando y muestran diferentes gráficos junto con comentarios relacionados sobre los mismos.

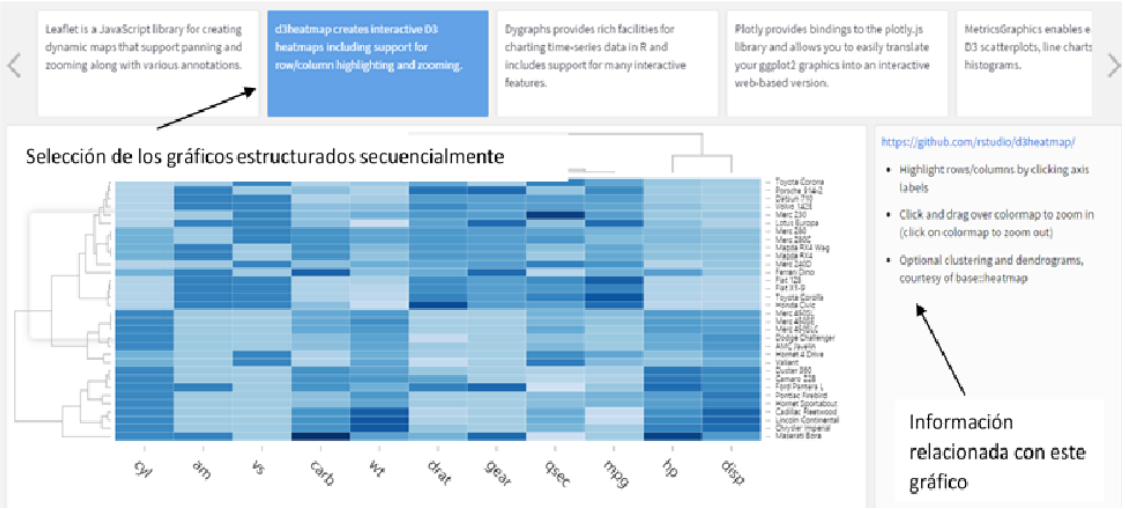

Figura 3.42: Ejemplo storyboard

- Por defecto, los **cuadro de mandos** son documentos *HTML* estándares que pueden ser implementados en cualquier servidor web o incluso adjuntarlas a un correo electrónico.
- Opción de añadir componentes *Shiny* para añadir interacción con el usuario, permitiendo, además, *"subirlo"* a un servidor (*Shiny Server* o *Shinyapps.io*).

3.6.2. Estructura básica

Flexdashboard está construido para ser utilizado a través de documentos **Rmarkdown**. Para generar este tipo de documento, simplemente, abrimos un documento **RMarkdown** y especificamos la opción *From template: Flex Dashboard*.

La estructura básica, que determina la salida visual del *cuadro de mando*, es muy fácil y limpia de escribir. La salida (*layout*) está determinadas por filas y columnas. Se determina, en una primera instancia, la columna y, dentro de ella, las filas que la van a componer.

Las secciones del **cuadro de mandos** vienen determinadas por un nivel 3 de cabecera de **RMarkdown** (###).

1) *Las columnas* se determinan a través del comando *"————————"*
2) *Las filas*, dentro de la columna se determinan, incluyendo diferentes *chunk* (trozos de código). Se estructurarán verticalmente dentro de cada columna

Veamos el siguiente ejemplo:

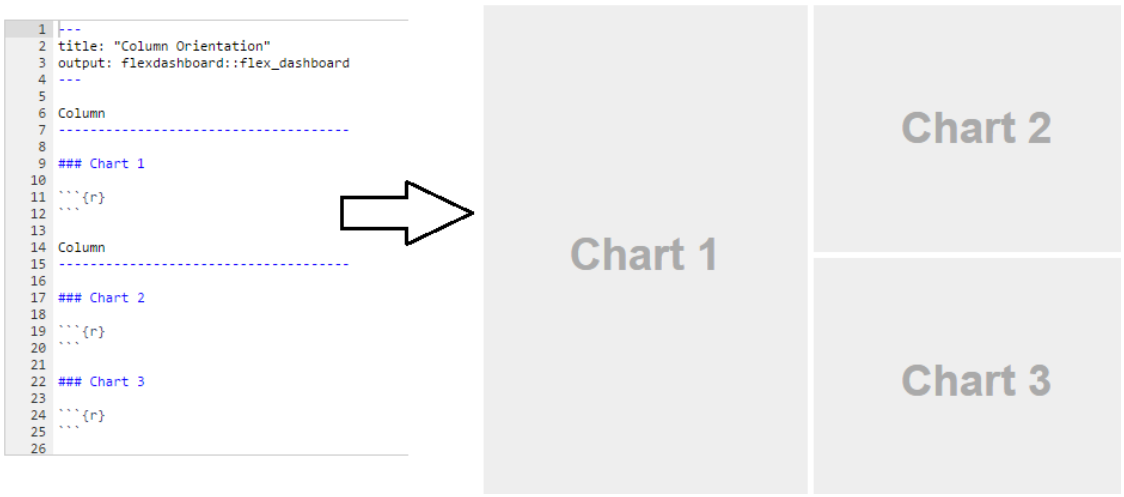

Figura 3.43: Código Flexdashboard y visualización

De esta forma, obtendríamos un *cuadro de mando* como el de figura 3.43, con dos columnas, con un gráfico u objeto a la izquierda y dos a la derecha. Como vemos presenta una manera muy sencilla de generar la estructura.

También se puede determinar la estructura especificando primero las filas, como podemos ver a continuación:

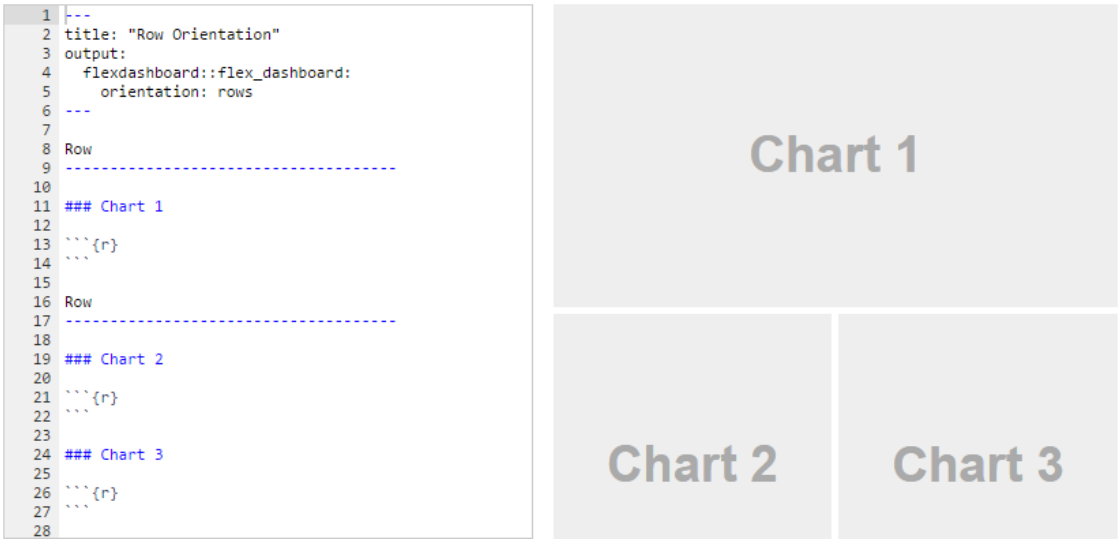

Figura 3.44: Elegimos la orientación del cuadro de mandos por filas a través del argumento orientation:rows, en la cabecera

3.6.3. Componentes

El paquete **flexdashboard** permite utilizar una serie de componentes que potencian y mejoran la visualización, dinamización e interactividad del *cuadro de mandos* que queramos construir.

3.6.3.1. HTML Widgets

Como hemos visto anteriormente, el paquete **htmlwidgets** utiliza una infraestructura *JavaScript* que otroga a R de una potente herramienta de visualización a partir de una serie de librerías.

Los gráficos basados en la estructura de **htmlwidgets** son ideales para utilizarlos conjuntamente con la estructura **flexdashboard**. Estos, pueden reajustarse de manera dinámica por si solos, de manera que la mayoría de las veces el resultado de la visualización estará perfectamente ajustado a la forma que delimitemos mediante la estructura **flexdashboard**

Como ya sabemos, el paquete **htmlwidgets** dispone de una extensa cantidad de paquetes widgets: *Plotly*, *Leaflet*, *dygraphs*, etc. Estos *htmlwidgets* se incluyen en el documento *RMarkdown* donde estamos generando nuestro *flexdashboard* como si fuera un gráfico normal de R.

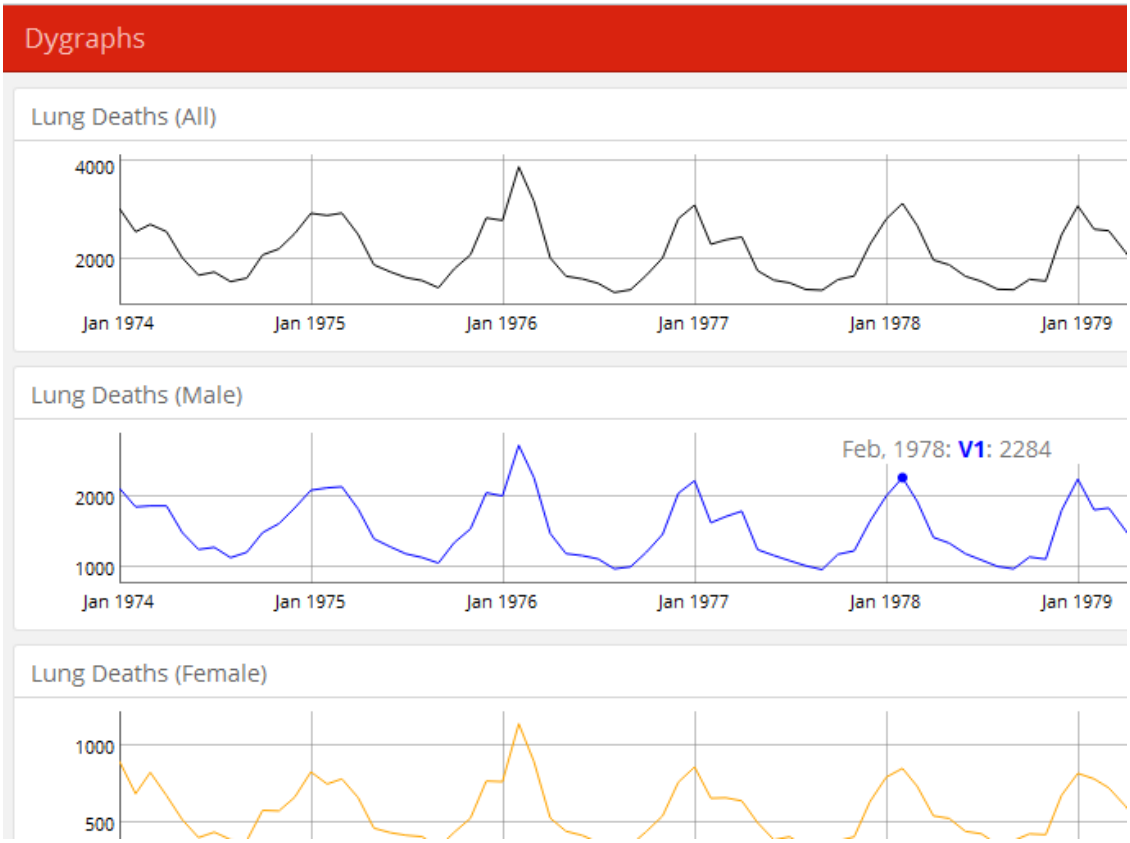

Figura 3.45: Elemento dygraphs en flexdashboard

Notemos, que **htmlwidgets** presenta una desventaja, y es que integra los datos directamente en la página web del servidor, por lo que se pueden encontrar problemas de rendimiento para grandes conjuntos de datos. Para esos casos, es preferible utilizar los componentes estándar que presenta **R Graphics**, que vamos a describir a continuación.

3.6.3.2. R Graphics

Utilizando la estructura **flexdashboard** podemos usar cualquier tipo de gráfica estándar que provee R en sus paquetes: *base*, *lattice*, *grid*, etc.

Si utilizamos **Shiny** para obtener un **cuadro de mandos** dinámico, esas gráficas se ajustarán automáticamente a la forma del *cuadro de mandos*

Si utilizamos las gráficas de manera estática, hay formas de ajustar el tamaño de estas de manera manual a través de los argumentos *fig.width* y *fig.height* que se insertan en las opciones del *chunk* en *RMarkdown*.

3.6.3.3. Tabular Data

Podemos incluir tablas dentro de nuestro **flexdashboard**, ya sea mediante una visualización normal y simple, o a través de un * DataTable* que permite ordenar, filtrar y paginar la tabla.

- Las tablas simples son especialmente útiles si estamos tratando con un número pequeño de casos (alrededor de 40 ó 50).
- El paquete **DT**, paquete está considerado dentro de **htmlwidgets** mencionado anteriormente.

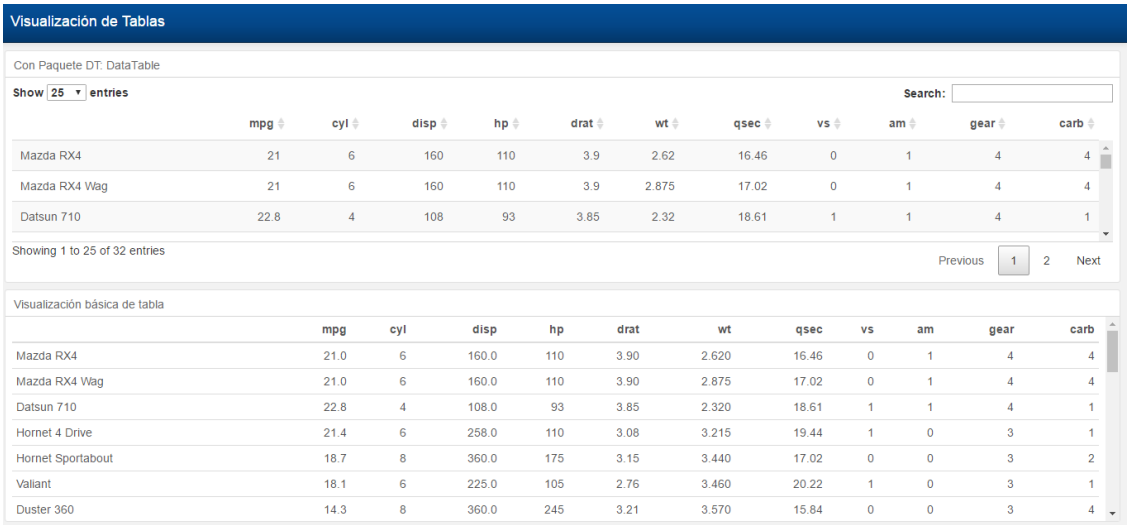

Figura 3.46: Visualización de tablas mediante paquete DT y de forma simple

3.6.3.4. Cajas de Valor

En algunas ocasiones puede ser interesante incluir en nuestro *flexdashboard* uno o más valores sencillos que muestren algún tipo de información que nos interese.

Esto, lo realizamos a través de la función *valueBox()*, incluida en el paquete **flexdashboard**, que permite mostrar un valor, junto con un título que lo describa y un icono opcional.

En el siguiente ejemplo (figura 3.47), vamos a ver tres cajas de valor:

- *Primera*: Donde se determina la fecha
- *Segunda*: Donde se determina la probabilidad de lluvia. Si la probabilidad de lluvia es mayor que 40 el icono es una nube, en caso contrario un sol.
- *Tercera*: Donde se determina la temperatura. Si la temperatura es superior a 20 ºC el color del fondo es naranja, en caso contrario celeste.

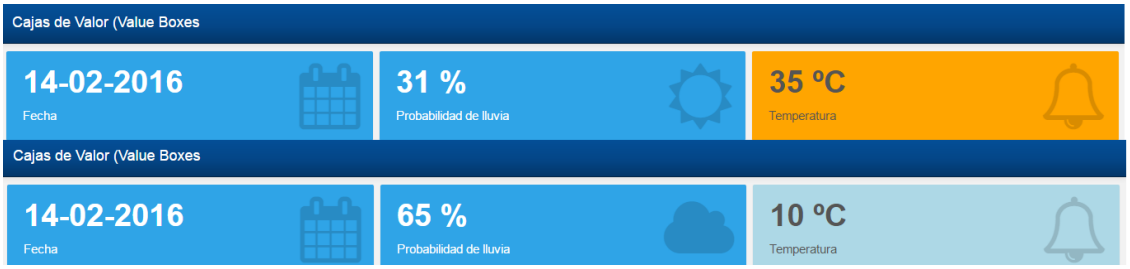

Figura 3.47: Ejemplo de Value Boxes

La función *valueBox()* da una opción muy interesante, y es que permite incluir una *referencia* en una caja de valor. Es decir, que al pulsar la determinada caja nos redirecciona a un link web, o bien a otro aparatado del *flexdashboard*. Esto se realiza a través del argumento *href* = "*http://...*"

3.6.3.5. Gauges

Los **gauges** muestran un valor a través de una especie de medidor dado un rango de valores especificados.

Esto se realiza a través de la función *gauge()* incluida en el paquete **flexdashboard**. Funciona a través del siguiente esquema:

- Le pasamos un valor numérico concreto.
- El valor mínimo y máximo del rango de valores donde está definido. Esto sirve para situar el valor.
- Argumento opcional para añadir un símbolo al lado del valor determinado (ej " %").
- Determinamos los sectores del **gauge**. Hay tres posibles sectores, donde a cada cual le corresponde un color: *success*(verde), *warning*(naranja) y *danger* (rojo). Determinamos que rango de valores corresponde a cada sector, de forma que según el valor se mostrará un color diferente.

Figura 3.48: Ejemplo de tres gauges en situación de éxito, precaución y peligro

Cabe destacar la gran importancia que puede tener este componente, *gauge*, a la hora del tema que nos atañe: *la inteligencia de negocios*. A partir de estas visualizaciones numéricas sencillas, que llevan asociados colores intuitivos sobre la situación, podemos obtener una idea rápida de los diferentes índices de una empresa y en el estado en el que se encuentran.

3.6.3.6. Anotaciones de Texto:

Si queremos introducir alguna información o explicación adicional en nuestro *cuadro de mandos* podemos hacerlo de dos maneras:

1. Podemos definir una sección del cuadro de mandos que no incluye gráficos pero que incluye contenido tipo texto, imágenes, ecuaciones, etc.

2. Introducir el contenido que queramos en la parte superior de la página antes de una sección (figura 3.49).

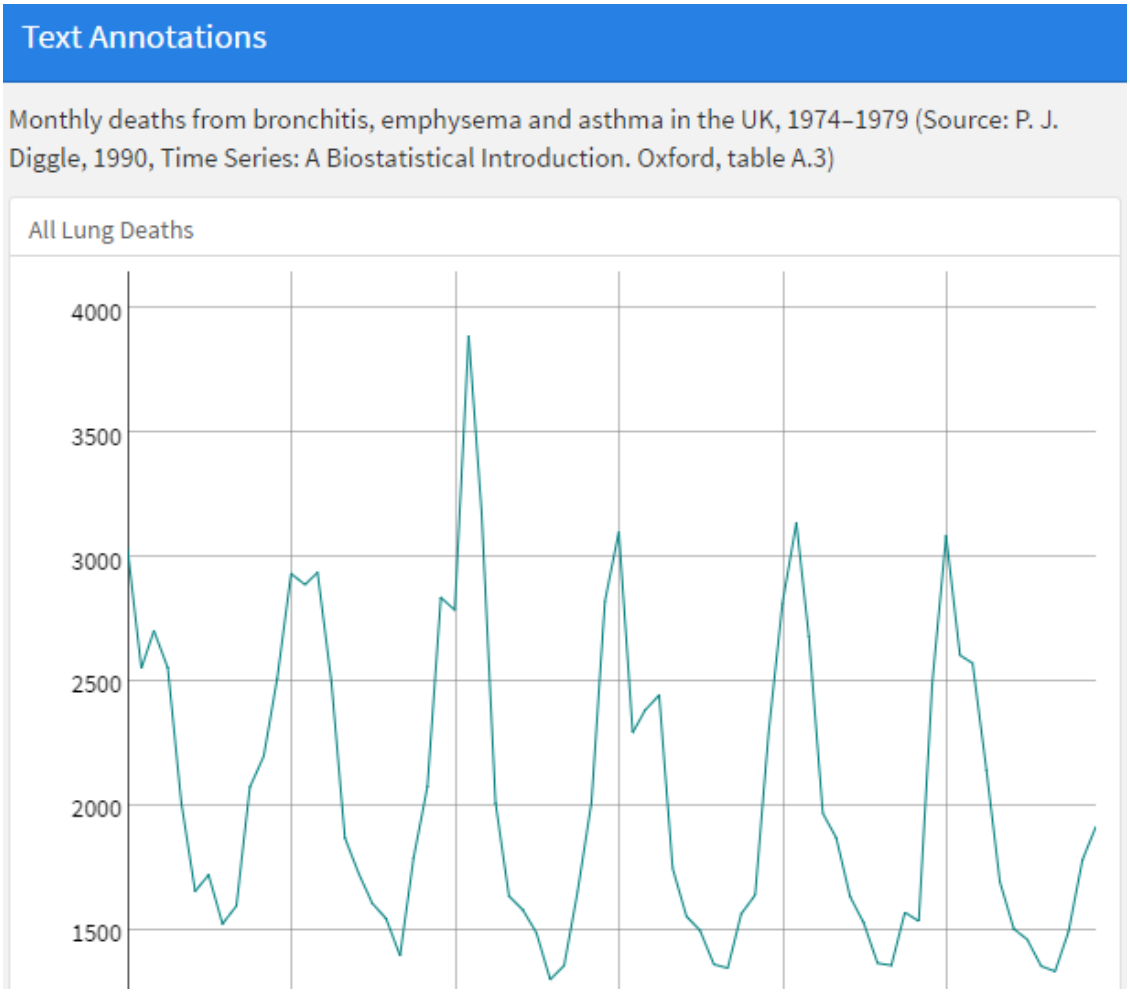

Figura 3.49: Información en la parte superior del cuadro de mandos

3.6.4. Más Opciones de visualización

El paquete **flexdashboard** presenta múltiples opciones de visualización y personalización. Vamos a presentar algunas de ellas:

3.6.4.1. Scrolling:

Por defecto, los gráficos u objetos con **flexdashboard** se establecen en la ventana del navegador de manera que se rellene la página al completo.

Este hecho es adecuado para cuando se tiene un número pequeño de gráficos u objetos colocados de manera vertical. Sin embargo, si empezamos a tener un número considerable de gráficos, posiblemente nos interese desplazarnos por la página en vez de encajarlos todos de manera estática en el navegador.

3.6.4.2. Pestañas:

Flexdahsboard permite utilizar pestañas seleccionables para visualizar diferentes gráficos u objetos. Esto lo realizamos a través de la opción *tabset*.

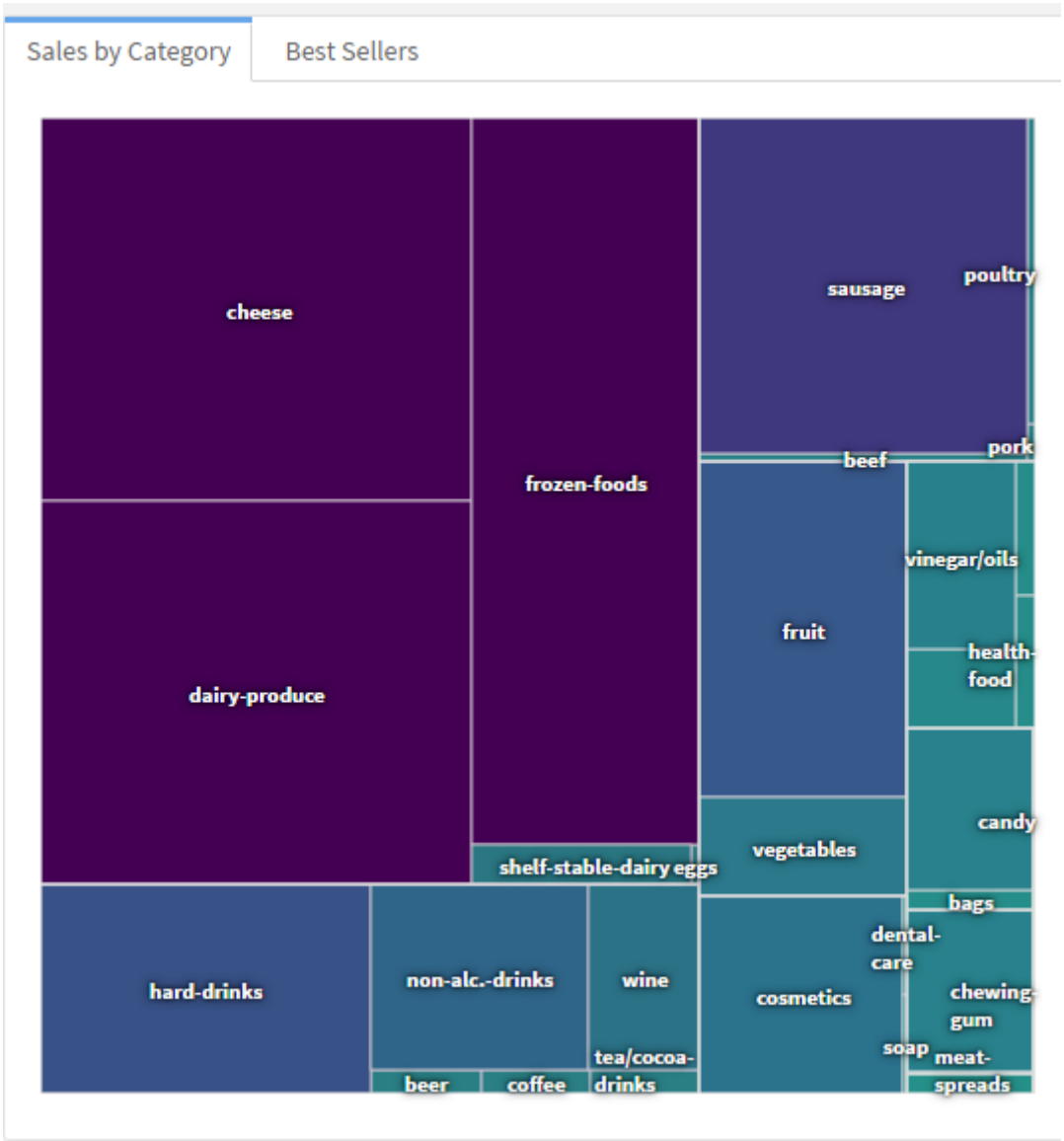

Figura 3.50: Ejemplo cuadro de mandos con dos pestañas

3.6.4.3. Sidebar:

Opción para mostrar una barra lateral con controles de opción sobre los gráficos. Esto lo realizamos a través de la opción *.sidebar*.

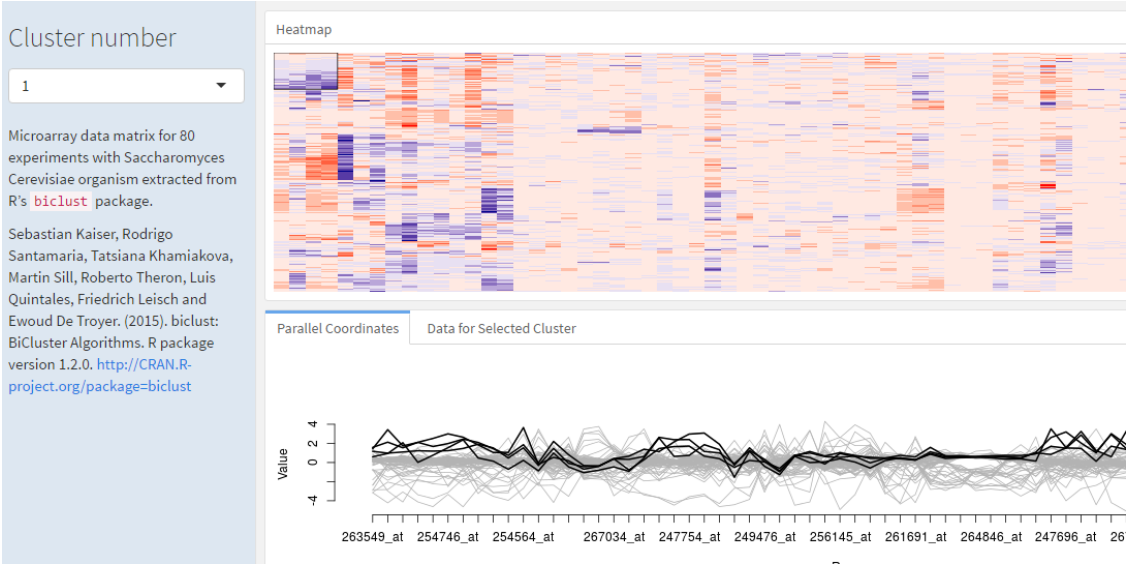

Figura 3.51: Barra lateral fija, sidebar.

3.6.4.4. Multi-Página

Mediante esta opción de salida, podemos construir nuestro **cuadro de mandos** a través de más de un página, simplemente utilizando (==========), como podemos ver en la figura 3.52.

```
1 ---
2 title: "Multiple Pages"
3 output: flexdashboard::flex_dashboard
4 ---
5
6 Page 1
7 ====================================
```

Figura 3.52: Código para delimitar una página: Título y ===

Cada página se podrá visualizar y seleccionar en la barra de navegación superior del **cuadro de mandos**. La estructura de cada página es independiente, es decir que, por ejemplo, en una página la orientación puede ser a través de columnas y la otra por filas.

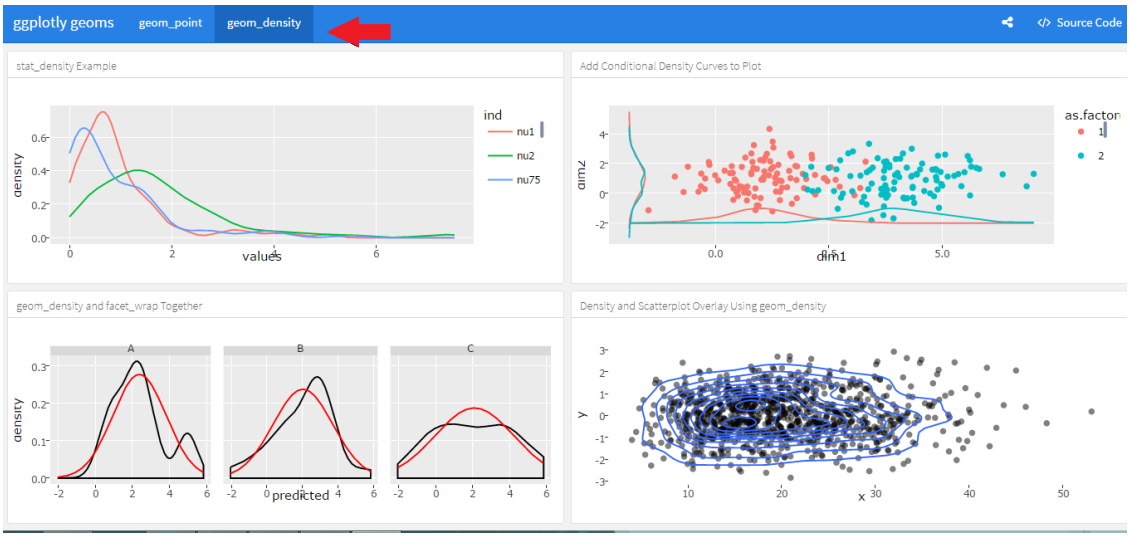

Figura 3.53: Ejemplo de cuadro de mandos con 2 páginas

Permite más opciones, como :

- Organizar las páginas por menús. Es decir, agrupándolas en diferentes categorías si se tienen muchas páginas.
- Hacer referencia de una página a otra, de manera que se crea un link que redirecciona a la página del **cuadro de mandos** que se decida.
- Permite añadir iconos a las pesteñas de las páginas que aparecen en la barra de navegación (figura 3.54). Dispone de una amplísima gama de iconos donde elegir. Podemos acceder a ellos desde los siguientes enlaces:

  a) Fontawesome: http://fontawesome.io/icons/

  b) Ionicons: http://ionicons.com/

  c) Glyphicons: https://getbootstrap.com/components/#glyphicons

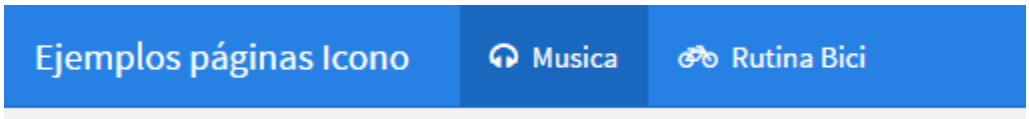

Figura 3.54: Ejemplo de páginas con iconos

3.6.4.5. Storyboards

Storyboards son una alternativa a la disposición o esquema de filas y columnas descrito anteriormente. Son usados para presentar un conjunto de gráficos u objetos de manera secuencial junto con comentarios relacionados para cada gráfico.

Para generarlo a través de **flexdashboard** basta con añadir la opción de *storyboard = true* en la cabecera del código e, incluir *### titulo* para determinar cada componente del cuadro de mandos.

Cada componente del **cuadro de mandos** es presentado con su correspondiente título en la barra de navegación superior.

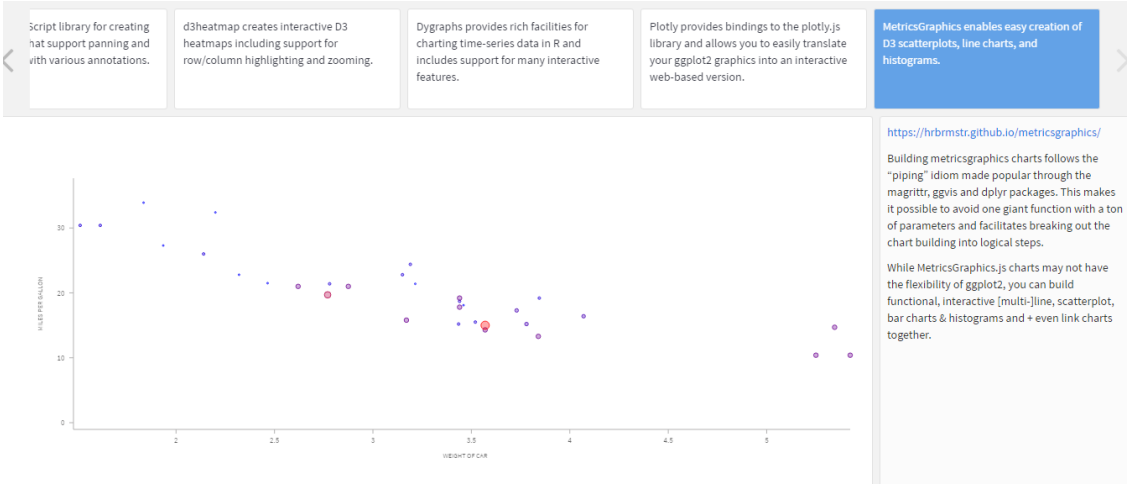

Figura 3.55: Ejemplo Storyboard con 5 secciones

También tenemos la opción de que sólo una página en concreto del cuadro de mandos presente la estructura de **storyboard**. Basta con añadir el argumento *{.storyboard}* a la página que queremos que tenga esa estructura.

3.6.4.6. Opción móvil:

Permite personalizar nuestro **cuadro de mandos** para una visualización en pantallas móviles. Realiza varias opciones sobre los gráficos u objetos que hemos definido para la visión web.

El paquete **flexdashboard** presenta un sistema inteligente que adapta automáticamente su salida de visualización a dispositivos móviles. El comportamiento que presenta dependerá del *tamaño de la pantalla*. El tamaño de las pantallas considerados es:

- Mediano: "Cualquier pantalla con un ancho superior a 768 píxeles", por ejemplo *tablets*. La visualización es igual que la obtenida mediante un ordenador
- Pequeño: "Cualquier pantalla con un ancho inferior o igual a 768 píxeles". Si la pantalla no dispone del ancho suficiente para mostrar diferentes columnas, todos los componentes del **cuadro de mandos** son dispuestos en una única columna. Podemos encontrar comportamientos especiales de adaptación a estos dispositivos en algunos elementos.

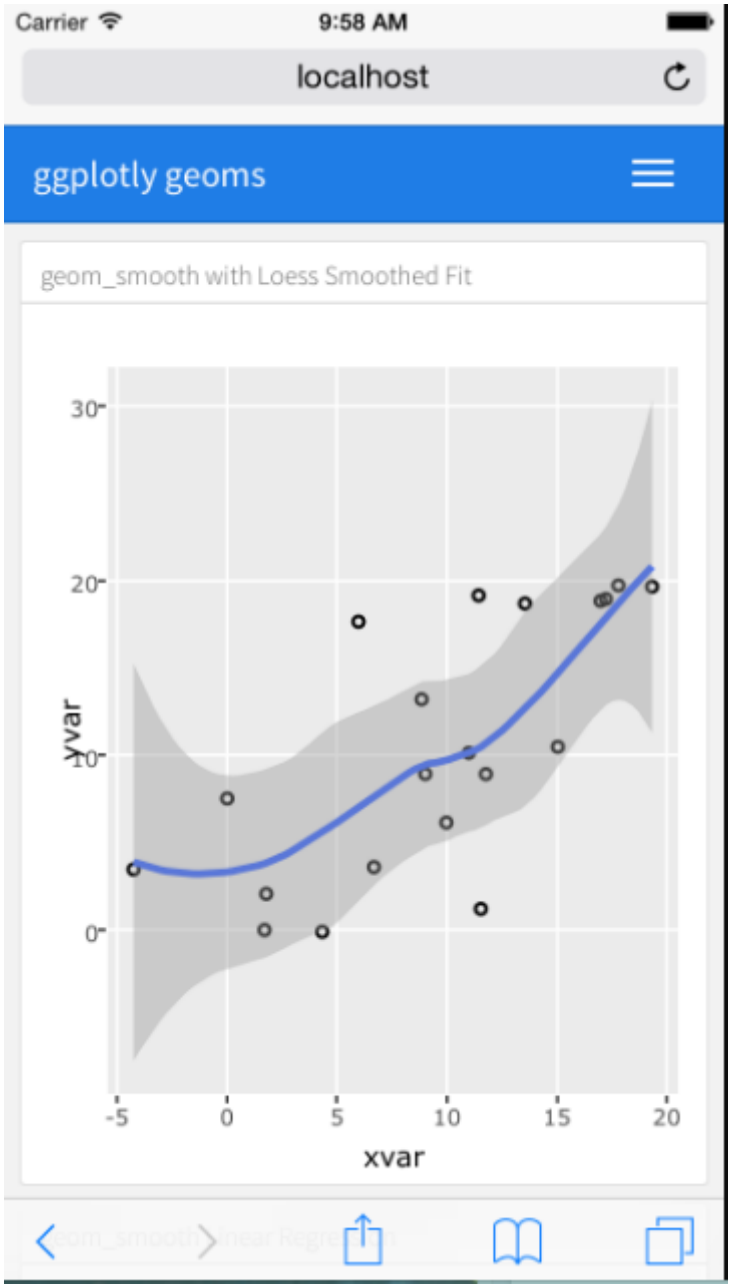

Figura 3.56: Visualización, en dispositivo móvil, de un cuadro de mandos realizado con paquete flexdashboard

Podemos seleccionar componentes que no queremos que aparezcan es caso de visualizar la aplicación a través de un formato móvil, o modificar algunos aspectos o características de los gráficos u objetos si estos son "abiertos" a través de un dispositivo móvil (figura 3.57).

Para excluir componentes basta con utilizar el comando *{.no-mobile}* en la especificación de su "título". En caso de querer visualizar un mismo gráfico, tanto en formato web como en dispositivo móvil, creamos dos títulos idénticos uno con el atributo *{.mobile}*

```
1  ---
2  title: "Mobile Specific"
3  output: flexdashboard::flex_dashboard
4  ---
5
6  ### Chart 1
7
8  ```{r}
9  plot(cars)
10 ```
11
12 ### Chart 2 {.no-mobile}
13
14 ```{r}
15 plot(pressure)
16 ```
17
18 ## Chart 3
19
20 ```{r}
21 plot(mtcars)
22 ```
23
24 ## Chart 3 {.mobile}
25
26 ```{r}
27 plot(mtcars)
28 ```
29
```

Figura 3.57: Ejemplo de código donde el chart1 es incluido tanto en salida web como móvil, el chart2 solo aparece en visualización web, y el chart3 aparece en ambas, con una personalización diferente para móvil

3.6.4.7. Otras opciones de apariencia:

Por defecto, el paquete **flexdashboard** presenta una amplia variedad de temas disponibles para modificar la apariencia básica del *cuadro de mandos*. Para especificar, si se quiere usar alguno, un tema basta con añadir la opción *theme: nombredeltema* en la cabecera del **flexdashboard** (figura 3.58).

```
1 ---
2 title: "Themes"
3 output:
4   flexdashboard::flex_dashboard:
5     theme: bootstrap
6 ---
7
```

Figura 3.58: Argumento para determinar el tema

Los temas que, por defecto, presenta el paquete **flexdashboard** son: *default*, *cosmo*, *bootstrap*, *cerulean*, *journal*, *flatly*, *readable*, *spacelab*, *united*, *lumen*, *paper*, *sandstone*, *simplex*, *yeti*.

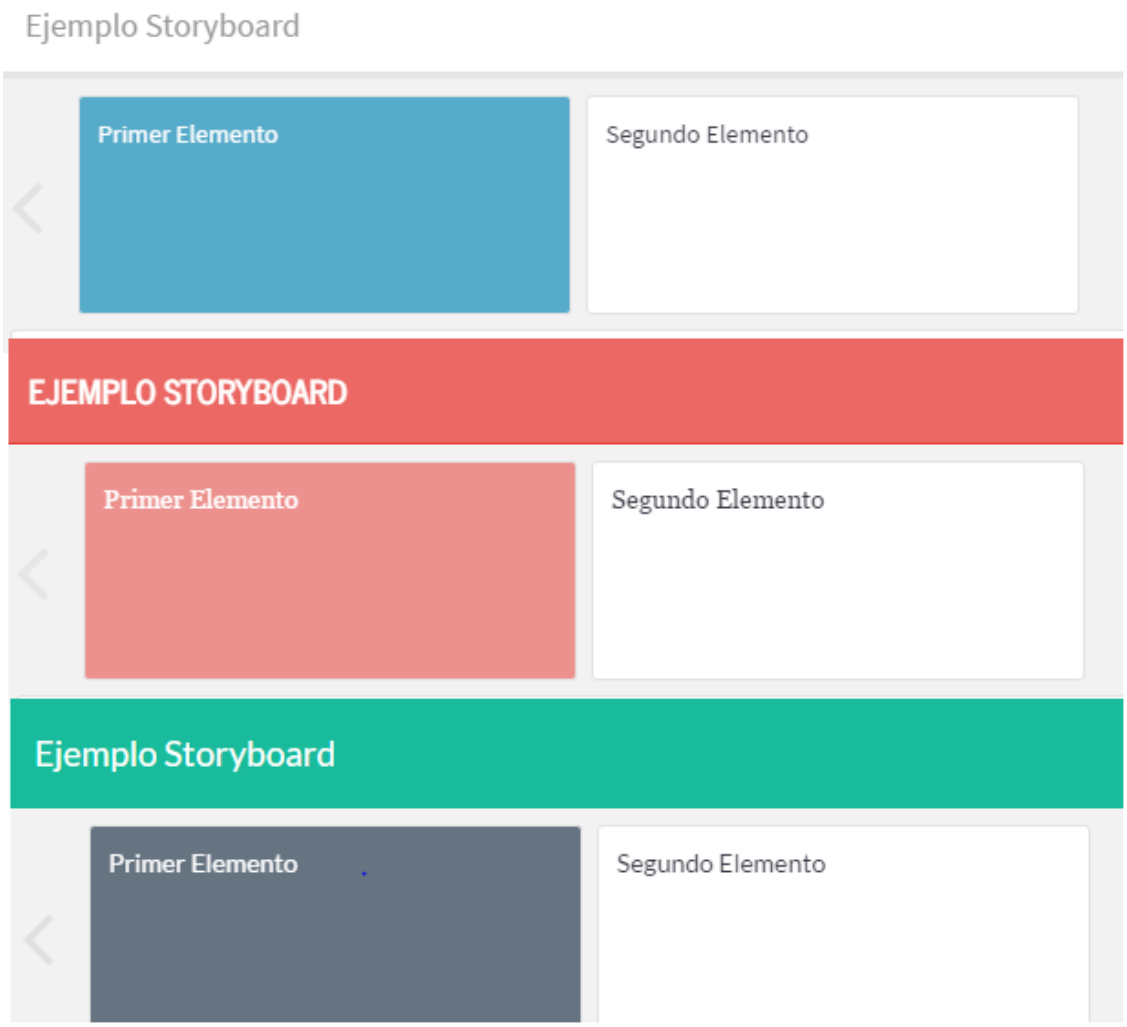

Figura 3.59: Ejemplo de temas: "lumen", "journal" "flatly", respectivamente.

Otra opción es añadir un fichero *CSS* de estilo para personalizar nosotros la apariencia del **cuadro de mandos**. Lo permite a través de la opción *css:*

Flexdashboard también permite incorporar un logo a nuestro **cuadro de mandos** (*figura 3.60: logo US*) para darle un toque más personal. Es importante mencionar que la imagen que pongamos como logo debe de tener el tamaño exacto para que ocupe el espacio que queramos, ya que no se autoajusta.

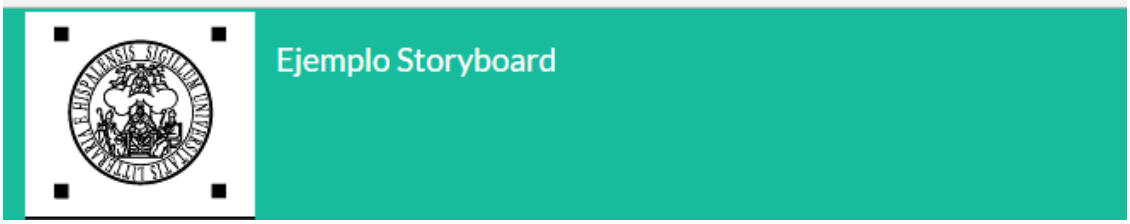

Figura 3.60: Flexdashboard con logo

3.6.5. Crosstalk: Gráficos interactivos

Crosstalk [14] es un complemento al paquete **htmlwidgets** que lo extiende a través de un conjunto de funciones y opciones que permite al usuario interactuar con los widgets. De esta manera, dejan de ser visualizaciones estáticas, pasando a tener opciones de filtro, actualización, selección etc.

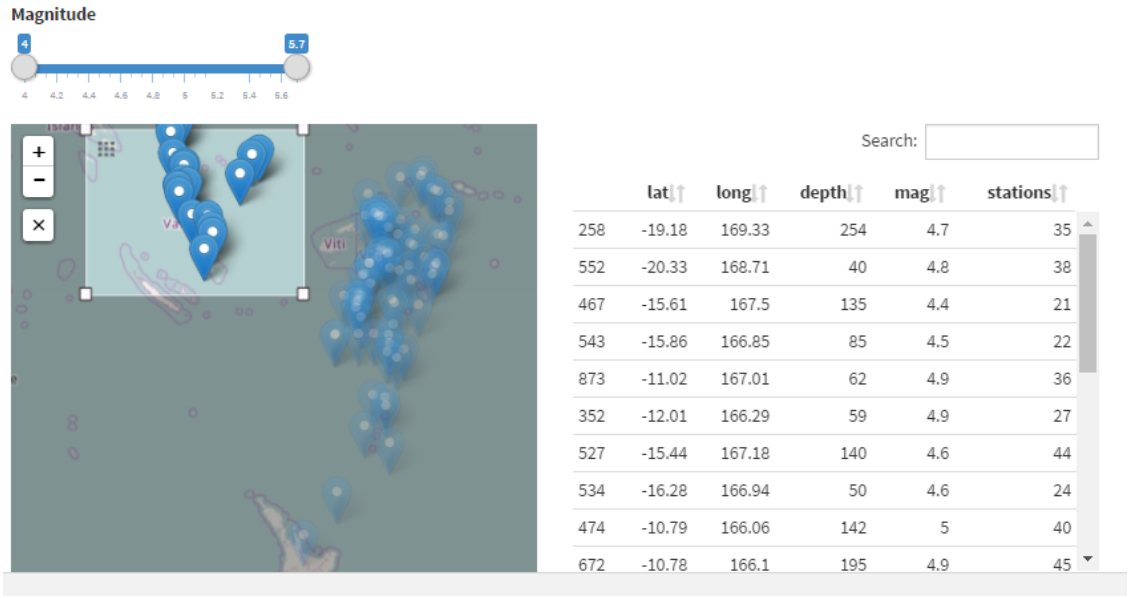

Figura 3.61: Ejemplo de widgets Plotly y DT con interacción por parte del usuario: Filtro de la variable mágnitud, y selección dentro del mapa las zonas que queremos mostrar en la tabla.

Al igual que cualquier otro *HTMLwidgets*, **Crosstalk** no requiere de **Shiny** ni ninguna otra dependencia de R para funcionar. Esto significa, que se puede usar "solo" o en un documento **RMarkdown** para generar documentos HTML estáticos que pueden ser subidos o alojados en cualquier sitio: *Amazon S3*, *GitHub* o *RPubs*.

Sin embargo, **Crosstalk** está diseñado para trabajar perfectamente con aplicaciones **Shiny**, o bajo la estructura **flexdashboard**, que te permiten combinar la interacción de los widgets Crosstalk con los gráficos básicos de R, los generados por **ggplot2**, etc. De esta manera, **Crosstalk** hace más fácil el poder unir múltiples *htmlwidgets* en un documento *RMarkdown* o en un aplicación **Shiny**.

3.6.5.1. Limitaciones

- No todos los widgets de **htmlwidgets** están preparados para la interacción con el usuario. Los propios autores de los *widgets* deben modificarlos para permitir el soporte de **Crosstalk**. **Crosstalk** no provee widgets interactivos a través de HTML widgets genéricos. Algunos de los *htmlwidgets* que son compatibles con **Crosstalk** son: *Plotly*, *Leaflet* o **DT**.

[14] https://rstudio.github.io/crosstalk/

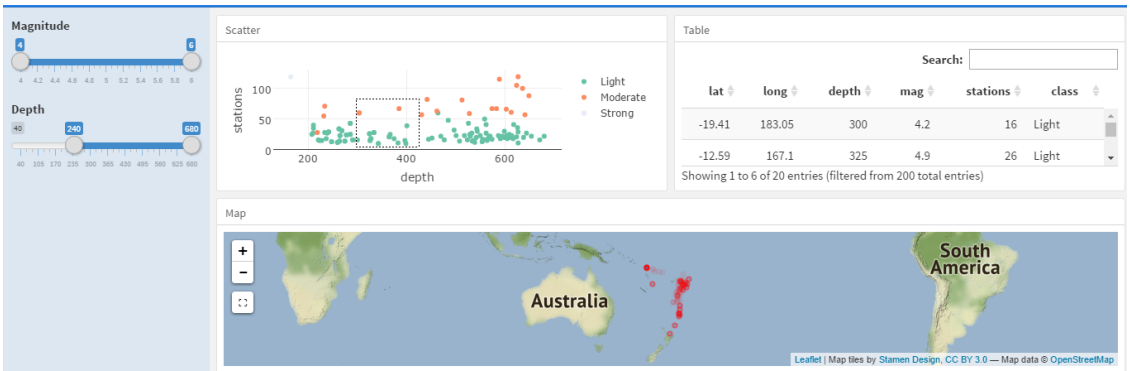

Figura 3.62: Crosstalk en conjucción con plotly (arriba izquierda), DT (arriba derecha) y Leaflet (abajo)

- Por ahora, **Crosstalk** solo presenta las opciones de filtrado en la visualización para puntos o datos individuales, y no presenta visualizaciones de datos agregados o resumidos. Por ejemplo, los histogramas no estarían soportados por **Crosstalk**, ya que cada barra representa un resumen de un conjunto de datos. Sin embargo un scatterplot representa un único punto y sí estaría permitido.
- El conjunto de datos completo debe de ser cargador en el navegador, por esa razón **Crosstalk no es apropiado para grandes conjuntos de datos**.

3.6.5.2. Algunos componentes

1. **Linked - Brushing**: Opción que permite una selección realizada con el ratón en el propio gráfico.

Esta opción resulta de más interés cuando tenemos un conjunto de datos con una serie de observaciones y representamos, mediante dos *widgets* diferentes, diferentes magnitudes.

De esta forma, a través de la selección en un gráfico de unos casos nos señalizaría en el otro gráfico esos mismo casos, que estarán en un orden y forma diferente.

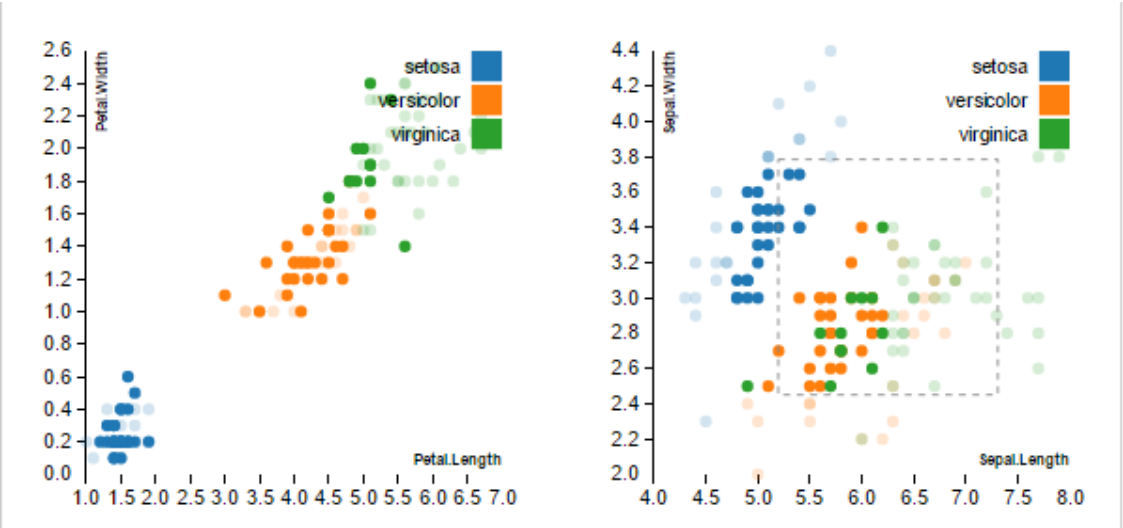

Figura 3.63: Ejemplo del uso de filtrado por marcado (brushing) usando crosstalk en widget de librería d3scatter

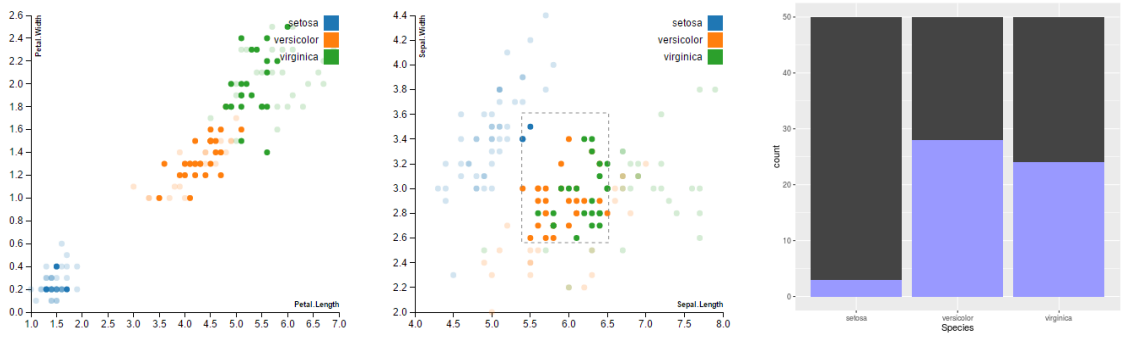

Figura 3.64: Ejemplo del uso de brushing con gráfico de barras creado con ggplot2 que muestra la cantidad de casos de cada especie seleccionadosen relacion con el total de cada especie.

2. **Filtrados *Inputs***: Opción de filtrar mediante la selección del usuario. Estos filtros pueden ser dados mediante selección de un conjunto de opciones, selección de valores en un rango determinado, o selección mediante lista despegable.

Podremos introducir más de un filtro a la vez, teniendo activos múltiples filtros (figura 3.65). De esta manera **Crosstalk** los combinará de forma que solo mostrará o resaltara aquellos datos que cumpla la intersección de todos los filtros activos.

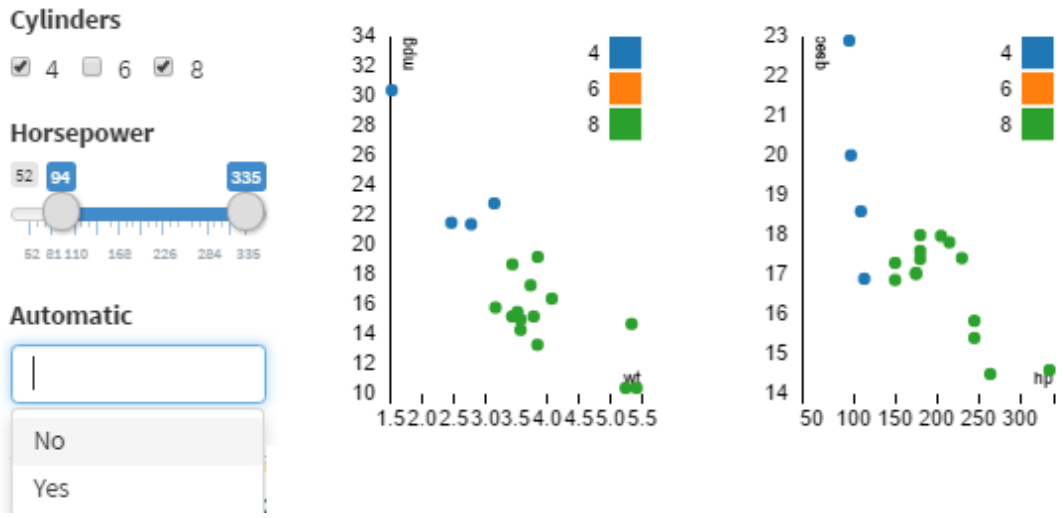

Figura 3.65: Ejemplo de 3 filtros diferentes activos usando Crosstalk y widget d3scatter

3.7. Comparación de R con otros Programas.

Como hemos visto a lo largo de este capítulo, a través de **R** y **RStudio** tenemos una amplia gama de posibilidades para realizar cuadros de mandos claros, visualmente llamativos, personalizables e interactivos. Por esta razón parece bastante adecuado y fiable considerar a **R** como una potente herramienta para el ámbito de la **inteligencia de negocios**, considerando, además, que se trata de un software libre y de muy fácil utilización, lo que lo provee de una gran ventaja sobre otros programas comerciales que hemos visto en el capítulo anterior.

Por lo tanto, llegados a este punto, nos preguntamos: *"¿Es R una posibilidad aceptable como una herramienta alternativa al uso de aplicaciones comerciales, como Tableau o Qlick, que están asentados con rigor desde hace tiempo en el mercado?"*

Para intentar dar una respuesta a esta cuestión, vamos a mostrar un ejemplo de un dahsboard producido por *Tableau* y el análogo obtenido a través de **R**. De esta forma tendremos una idea del alcance y limitaciones que presenta **R**.

3.7.1. Potencialidad de R en BI

Como ya hemos comentado anteriormente, una de las mayores ventajas que presenta **R** son los miles de paquetes que los usuarios han generado para resolver problemas específicos en distintas disciplinas, desde el análisis de datos meteorológicos o financieros hasta el modelado del genoma humano.

Independientemente del área al que estemos enfocándonos, algunas tareas son comunes a todas: La importación de datos, su lectura y discusión y, por último, su visualización.

Hemos visto que R presenta múltiples paquetes que dotan de unas visualizaciones claras y llamativas que permiten una lectura fiable y rápida de la información que se obtiene de los datos. Esto recordemos que es una cuestión fundamental en la **inteligencia de negocios** que se fundamenta en proporcionar herramientas intuitivas, fiables y actualizadas que permitan al usuario llegar a tener una visión real de la información para poder así tomar decisiones con seguridad.

3.7.2. Ejemplo

Podemos ver un ejemplo, creado por *Len Kiefer*, en el que se va a transformar, o más bien tratar de imitar, un *cuadro de mandos* generado por **Tableau** a través de la herramienta **flexdashboard** de **R**.

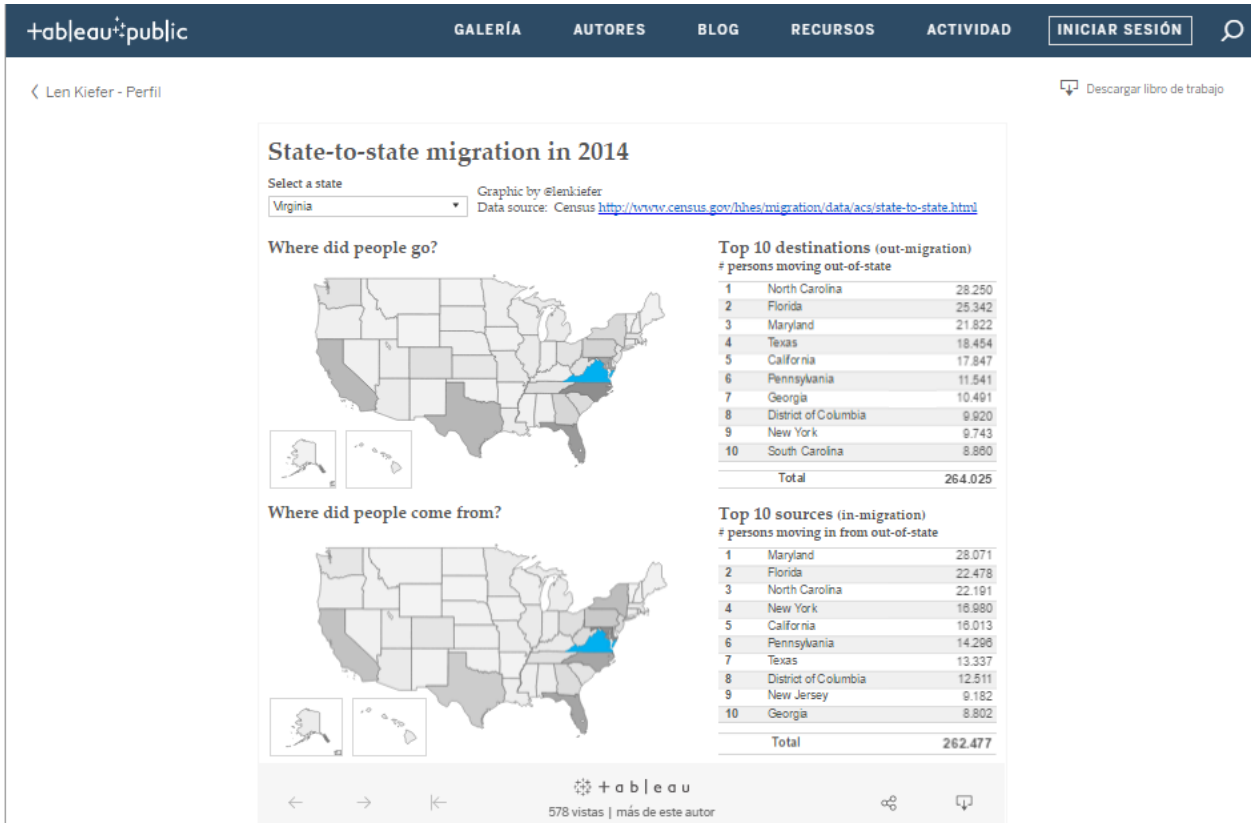

Figura 3.66: Cuadro de Mandos original con Tableau

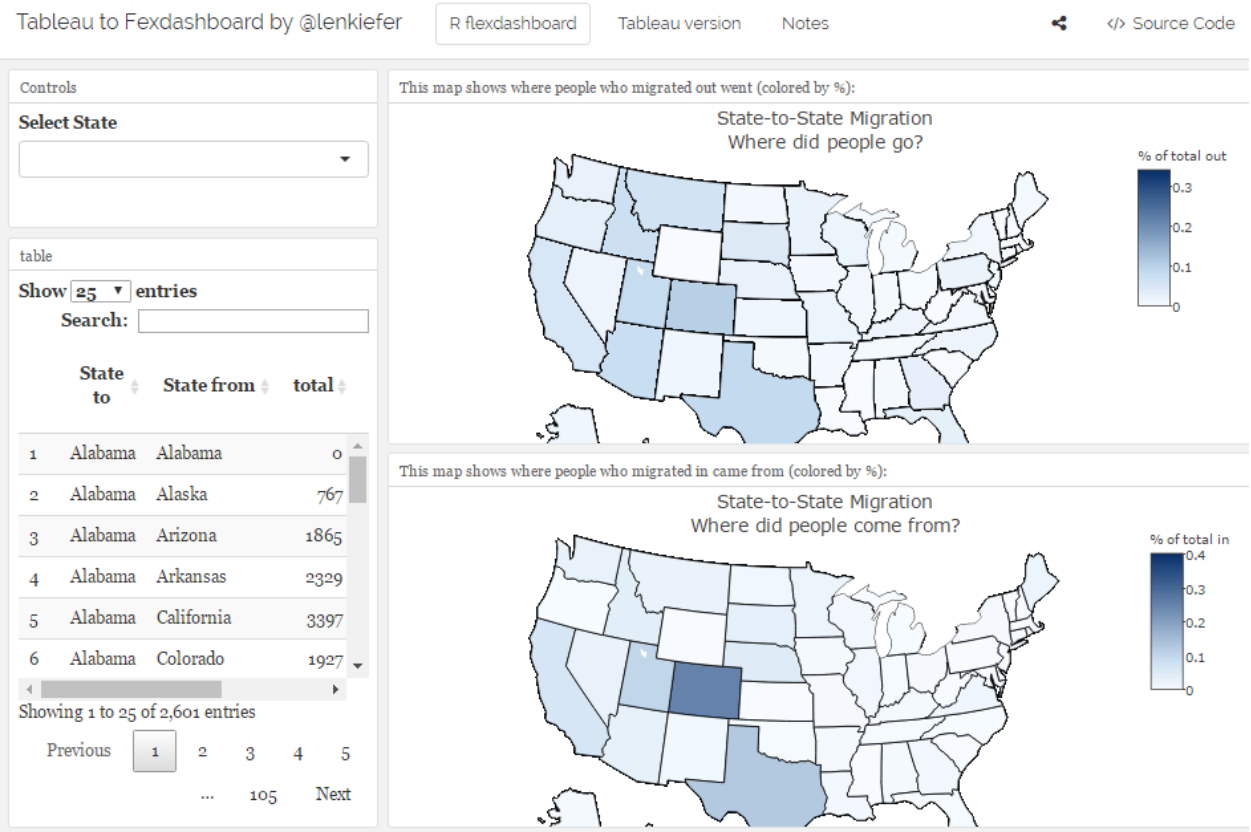

Figura 3.67: Cuadro de Mandos obtenido con R

Podemos ver el proceso completo y detallado en el la web oficial de Len Kiefer[15]

3.7.2.1. Conclusiones

A través de este ejemplo, hemos podido ver como el software libre de **R** nos puede proporcionar una herramienta potente para presentar **cuadros de mandos** que se asemejan, con bastante calidad, a los ofrecidos por las aplicaciones comerciales.

Además, también hemos visto la disposición de la amplia gama de paquetes que encontramos en R para poder realizar las diferentes fases de la inteligencia de negocios: extracción de datos, aplicación de data mining, opciones de visualización, etc.

[15] http://lenkiefer.com/2017/01/26/convert-tableau-to-r

Capítulo 4

Cuadro de Mandos con R

4.1. Objetivo

En este capitulo vamos a ejemplificar la capacidad que tiene **R** para crear potentes cuadros de mandos acercándose bastante a lo que ofrecen las aplicaciones comerciales que hemos presentado en capítulos anteriores. Vamos a mostrar algunas de las múltiples posibilidades que disponemos para presentar los datos y la información a través de R.

Para crear nuestro cuadro de mandos vamos a utilizar diferentes librerías de R y que nos permiten realizar los pasos necesarios para crear un buen cuadro de mandos:

1) Lectura de datos.
2) Tratar y preparar de los datos.
3) Análisis estadísticos.
4) Representación de los resultados.

4.2. Contexto

Con el cuadro de mandos que vamos a generar queremos visualizar diferente información referente a los alumnos adcritos a las diferentes titulaciones de la Facultad de Matemáticas. La información de la que partimos son el número de alumnos y su distribución según diferentes características: Titulación, Curso, Asignatura, etc..

A través de esta información queremos obtener un cuadro de mandos que muestre diferentes representaciones gráficas e indicadores que resulten de interés para la gestión, conocimiento y organización administrativa de la facultad, así como otros aspectos, como la evolución histórica de los datos, de manera que tengamos una idea de como ha ido avanzando y creciendo la facultad.

Es importante resaltar que los datos con los que trabajaremos, en algunos casos, no son reales. El objetivo de este capítulo es la generación del cuadro de mandos para ver como se puede mostrar la información relevante y de valor.

4.3. Los datos

Antes de plantear cómo vamos a generar el cuadro de mandos, vamos a hacer una presentación más detallada de la estructura de los datos originales de los que vamos a partir.

Tenemos el número de alumnos según las siguientes variables:

- Titulación: Grado en Matemáticas, Grado en Estadística, Doble Grado en Física y Matemáticas, Doble Grado en Matemáticas y Estadística y Máster Universitario en Matemáticas.
- Asignaturas: Asignaturas que se imparten en cada titulación.
- Identificador Asignatura.
- Duración: Se refiere a si la asignatura es Anual o del Primer o Segundo Cuatrimestre.
- Carácter: Si la asignatura es obligatoria o si es optativa.
- Créditos: Créditos de la asignatura. Pueden ser 12, si es anual, o 6, si es cuatrimestral.
- Curso: Para los grados hay 4 cursos, para los dobles grados hay 5 cursos y el máster es un único curso.
- Alumnos 2012/2013: Número de alumnos matriculados para el curso académico 2012/2013.
- Alumnos 2013/2014: Número de alumnos matriculados para el curso académico 2013/2014.
- Alumnos 2014/2015: Número de alumnos matriculados para el curso académico 2014/2015.
- Alumnos 2015/2016: Número de alumnos matriculados para el curso académico 2015/2016.
- Alumnos 2016/2017: Número de alumnos matriculados para el curso académico 2016/2017.
- Repetidores: Número de alumnos repetidores referentes al curso académico 2016/2017.
- Porcentaje repetidores: Porcentaje de alumnos repetidores referentes al curso académico 2016/2017.
- Grupos: Referente al número de grupos que hay para cada asignatura en el curso académico 2016/2017.
- Media por grupos: Número medio de alumnos po cada grupo de una asignatura.
- Transversal: TR si la asignatura es transversal.
- Departamento: Departamento al que está adscrito la asignatura
- Área: Área a la que está adscrita la asignatura
- Nº Departamentos: Número de departamentos que se encargan de impartir la asignatura.

Principalmente, las variables con las que vamos a trabajar son las siguientes: Titulación, Curso, Asignatura, Alumnos por curso académico y repetidores.

4.4. Proceso hasta el cuadro de mandos con R.

Listado de librerías adicionales a las básicas implementadas en R que vamos a utilizar:

1. **Lectura de datos**: readxl.
2. **Manipulación de datos**: dplyr.
3. **Representaciones gráficas y cuadro de mandos**: ggplot2, gridExtra, DT, plotly, leaflet, dygraphs, flexdashboard, scatterD3, highcharter, crosstalk y shiny. Para poder utilizar correctamente algunos de estos paquetes también usaremos xts y zoo, para la representación de series temporales.

4.4.1. Lectura de los datos

El primer paso para poder empezar a trabajar, manipular y transformar los datos para llegar a nuestro objetivo final, es leer los datos.

Tenemos 5 conjuntos de datos, uno para cada titulación nombrada. Cada conjunto de datos está guardado en un formato *xlsx*. Podemos ver cual es la estrucutra de los archivos Excel en la figura 4.1

	B	C	D	E	F	G	H	I	J	K
1	idtit	Titulación	idasig	Asignatura	curso	duracion	caracter	creditos	Alum 12/13	Alum 13/14
2	196	Grado en Estadística	1960001	Algebra Lineal	1	A	O	12	40	35
3	196	Grado en Estadística	1960002	Cálculo	1	A	O	12	42	40
4	196	Grado en Estadística	1960004	Cálculo de Probabilidades	1	C1	O	6	36	48
5	196	Grado en Estadística	1960005	Estadística Descriptiva	1	C1	O	6	37	39
6	196	Grado en Estadística	1960003	Informática	1	A	O	12	31	36
7	196	Grado en Estadística	1960006	Programación Matemática	1	C2	O	6	39	44
8	196	Grado en Estadística	1960007	Teoría de la Probabilidad I	1	C2	O	6	45	49

Figura 4.1: Estructura fichero de datos en Excel.

Para la lectura de los datos, al estar en este tipo de ficheros con extensión *xlsx*, utilizamos la librería **readxl** y, en concreto, la función **read_excel** de dicho paquete. De esta manera ya tenemos cargados los datos en nuestro entorno de R.

4.4.2. Manipulación de los datos

Una vez que disponemos de los datos ya cargados ya podemos modificarlos para que estén en el orden y en el formato que nosotros consideremos más conveniente para el trabajo que queremos abordar.

Tras la lectura de los datos, tenemos 5 conjuntos de datos en formato *data_frame*, uno para cada titulación. Lo que nos interesa es lo siguiente:

1. Tener un único conjunto de datos donde aparezcan los datos para todas las titulaciones. Para ello basta con aunar los 5 conjuntos de datos con la función **rbind** que añade por filas cada conjunto de datos.
2. Quedarnos con las variables que nos resultan de interés para el trabajo. Basta con seleccionar los las columnas que representan las variables que nos interesan: Centro, Titulación, Asignatura, Curso, Alum12_13, Alum13_14, Alum14_15, Alum15_16, Alum16_17, repe, tantorepe, Grupos16_17, Activ y DptoPral. Para ello utilizamos la función **select** del paquete **dplyr**.
3. Correcto formato de las variables. Definir como *factor* las variables que toman valores categóricos.

Una vez que tenemos el conjunto de datos tal como nos interesa, vamos a obtener información a partir de él

1. Número total de matriculados en cada curso académico. Utilizamos la función *select* y *summarise* del paquete **dplyr**. Con la función **summarise** calculamos un valor a partir de realizar alguna operación sobre las columnas que queramos de un conjunto de datos, en este caso la suma para las variables: "Alum12_13", "Alum13_14", "Alum14_15", "Alum15_16", "Alum16_17". Podemos ver su utilización en el siguiente recorte de código.

```
total_matriculados_ano = datos %>%
  select(Alum12_13:Alum16_17) %>%
  summarise(Total12_13 = sum(Alum12_13), Total13_14 = sum(Alum13_14),
            Total14_15 = sum(Alum14_15),
            Total15_16 = sum(Alum15_16), Total16_17 = sum(Alum16_17, na.rm = T))
```

2. Número total de matriculados por curso académico y títulación. Utilizamos las mismas funciones que en el caso anterior, pero agrupando por títulación, utilizando la función *group_by* de **dplyr**. De manera

que tendremos el total de alumnos de cada curso académico para cada títulación. Obtendremos un conjunto de datos tal como el que podemos ver en la figura 4.2

```
# A tibble: 5 × 6
                                Titulacion Total12_13 Total13_14 Total14_15 Total15_16 Total16_17
                                    <fctr>      <dbl>      <dbl>      <dbl>      <dbl>      <dbl>
1      Doble Grado en Física y Matemáticas          0        105        285        466        675
2 Doble Grado en Matemáticas y Estadística          0         80        232        400        650
3                    Grado en Estadística        491        738        793        853        869
4                    Grado en Matemáticas       3341       3802       3958       4100       4060
5     Máster Universitario en Matemáticas          0          0         73          0        179
```

Figura 4.2: Total matriculados por curso académico y titulación.

3. A partir del número de matriculados por curso académico y por titulación, y teniendo también el número total de matriculados en cada curso, podemos obtener el porcentaje de matriculados que representa cada titulación. De manera que podremos ver como se distribuyen los alumnos por titulación y curso.

A partir de esta información generada vamos a ir construyendo nuestro cuadro de mandos.

4.4.3. Generación del cuadro de mandos

Para la creación de nuestro cuadro de mandos con R vamos a basarnos principalmente en las opciones que nos presenta la librería **flexdashboard**. A través de ella, vamos a construir la estructura y el esqueleto de nuestro cuadro de mandos que iremos rellenando con diferentes representaciones gráficas e indicadores.

Antes de iniciar los pasos que vamos a realizar, tenemos que indicar en nuestro archivo **.Rmd** que lo que vamos a generar es un cuadro de mandos con **flexdashboard**. El siguiente código lo introducimos al inicio del código (script).

```
---
title: "Datos Facultad"
output:
  flexdashboard::flex_dashboard:
     theme: sandstone
     logo: logom.PNG
     source_code: embed
     social: [ "twitter", "facebook", "menu" ]
---
```

Con este código también determinamos lo siguiente:

- Tema visual que se va a utilizar en el cuadro de mandos (theme: sandstone).
- Logo o icono del cuadro de mandos (logo: logom.PNG).
- Opción de que el código esté disponible en el cuadro de mandos (source_code: embed).
- Iconos de redes sociales para compartir a través de ellas el cuadro de mandos si se desea (social: [“twitter”, “facebook”, “menu”]).

Figura 4.3: Encabezado dashboard.

4.4.3.1. ÍNDICE DE LA PÁGINA

Al entrar en nuestro cuadro de mandos, lo primero que vemos es un resumen de la información recogida a través de un índice. También vemos un pequeño resumen y presentación sobre el fin de generar el cuadro de mandos, junto con un mapa donde aparece la localización de la facultad de matemáticas.

- El índice aparece en una barra lateal a la izquierda. Presenta opción de *click* en cada link que redirecciona a cada apartado comprendido en diferentes páginas. Estas opciones las construimos con la opción *sidebar* de **flexdashboard** para colocarla en un lateral y ponemos entre corchetes los nombres de cada página para que al seleccionarlas enlace con ellas.

```
Introducción
====================================

Inputs {.sidebar}
------------------------------------

Dashboard creado por MA MORA MAQUEDA

[Titulación]

[Curso y Titulación]

[Índices Informativos]

[Con Plotly]

[Con Dygraph]

[Comparación Total Matriculados]

[Matriculados y Repetidores]

[Repetidores Matemáticas]

[Repetidores Estadística]
```

Figura 4.4: Índice con enlaces.

- El mapa es generado a través de la librería **leaflet**, indicando con un marcador la posición donde se encuentra la Facultad de Matemáticas.

```
leaflet() %>%
  addTiles() %>%
  addMarkers(lng=-5.98814, lat=37.35945, popup="Facultad de Matemáticas")%>%
  setView(lng=-5.98814, lat=37.35945, zoom = 18)
```

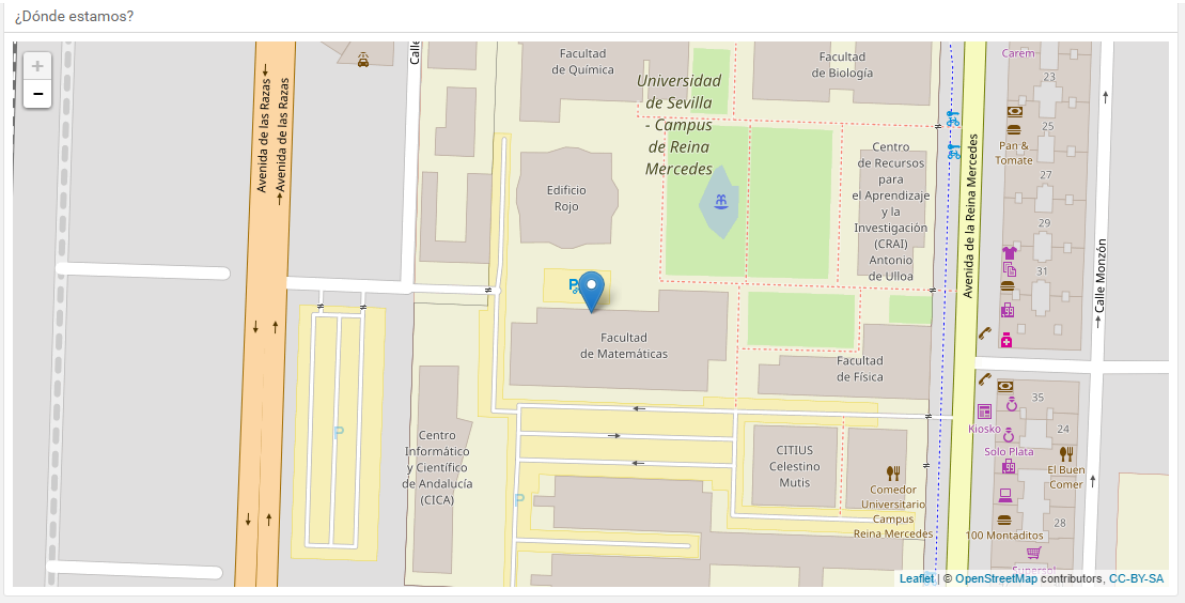

Figura 4.5: Mapa con localización Facultad Matemáticas

4.4.3.2. PORCENTAJES

Dentro de la pestaña **porcentajes** encontramos dos páginas: *Titulación* y *Curso y Titulación*.

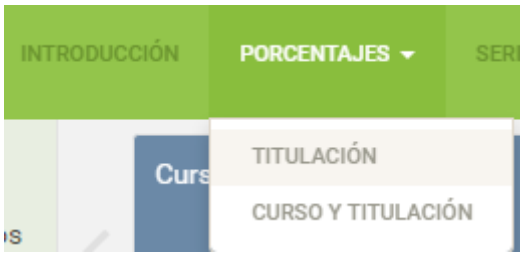

Figura 4.6: Páginas con menú desplegable

Esta estructura, en la que en la barra de navegación encontramos las dos páginas desglosadas dentro de un mismo título *Porcentajes*, lo realizamos a través del siguiente código; Donde por un lado decimos el nombre de la página "individual" y, entre corchetes, especificamos la etiqueta que va a englobar a esa página.

```
Titulación {.storyboard data-navmenu = "Porcentajes"}
======================================
```

De igual forma, para *Curso y Titulación* con *data-navmenu = "Porcentajes"*.

El atributo *.storyboard*, como ya explicamos en el capítulo anterior, nos permite tener una estructura organizada mediante cuadros "pulsables", donde quedan recogidos diferentes gráficos junto con la posibilidad de añadir un texto descriptivo a cada uno. De esta forma obtenemos, por ejemplo, la siguiente visualización para el caso de *Títulación*.

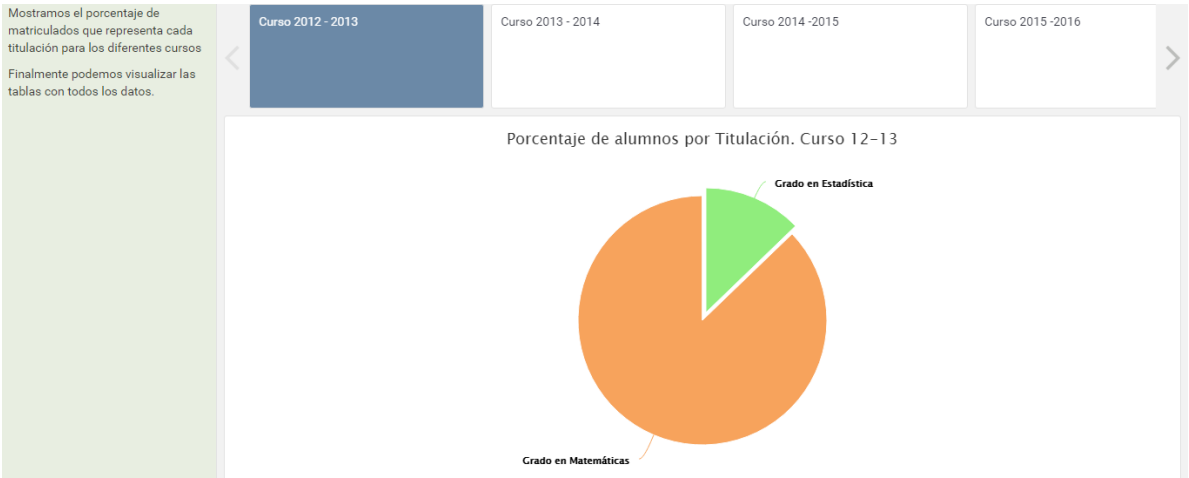

Figura 4.7: Estructura de storyboard

TITULACIÓN

En este apartado del cuadro de mandos queremos tener una visión sobre el porcentaje de alumnos que representa cada titulación sobre el total de alumnos de la Facultad de Matemáticas. Para ello, mediante la estructura de *storyboard* que acabamos de mostrar, diferenciamos para cada curso académico. De esta forma podemos ir "pinchando" en los diferentes cuadros que mostrarán los gráficos.

En esta página tenemos:

- Descripción general sobre lo que vamos a encontrar en ella (a través de *sidebar* como ya hemos visto). Esta información queda fija a medida que avanzamos en el storyboard.
- Diferentes cuadros de gráficos: Curso 2012-2013, Curso 2013-2014, Curso 2014-2015, Curso 2015-2016, Curso 2016-2017. En los gráficos encontramos diagramas de sectores, como el que podemos ver en la figura 4.8.

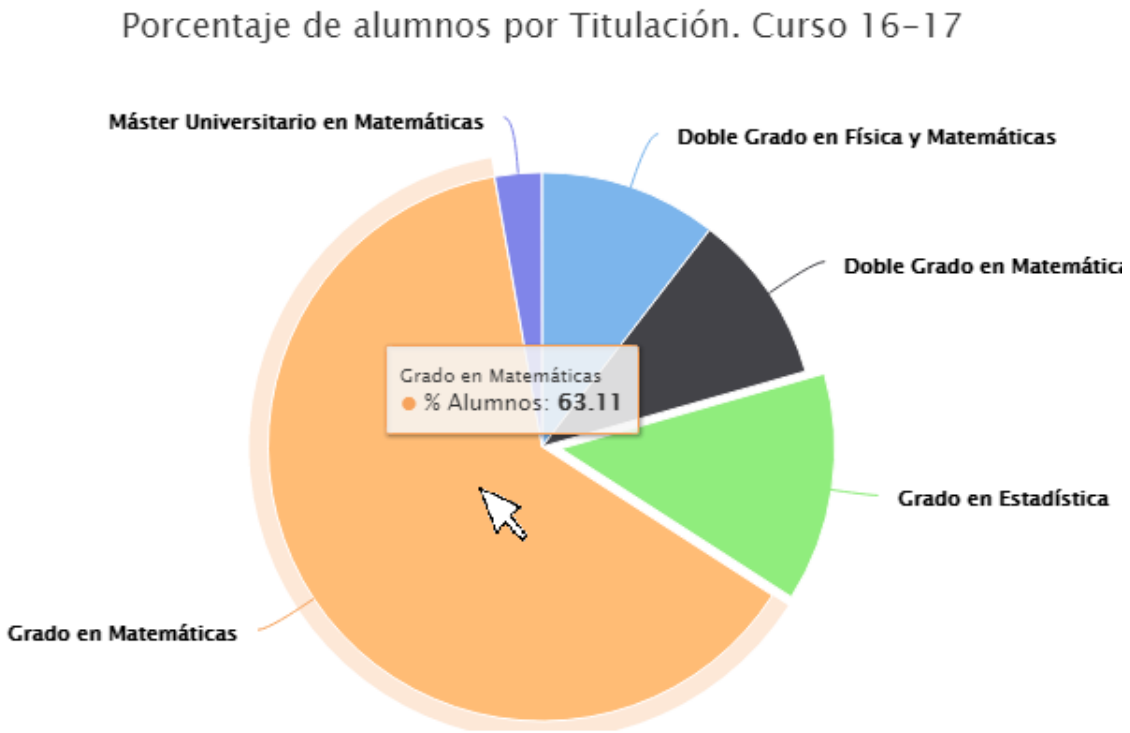

Figura 4.8: Porcentaje alumnos por titulación. Curso 16-17

Estos diagramas de sectores se realizan a través de la libreria **highchart**, que nos permite hacer diferentes tipos de gráfico, en este caso *type = pie*.

Se muestran los diferentes sectores que hacen referencia a las titulaciones. Al pasar el cursor por alguno de los sectores se muestra el dato concreto del porcentaje para esa titulación. Vemos también que se ha resaltado el sector referente a la titulación de *Estadística*, separándola del circulo central (*slice = TRUE*). Además, con el argumento *dataLabels = list(enabled = FALSE)*, podemos ocultar el nombre de la etiqueta para esa titulación. Vamos a ver el código referente a este gráfico para ver un ejemplo del uso de esta librería:

```
highchart() %>%
  hc_chart(type = "pie") %>%
  hc_title(text ="Porcentaje de alumnos por Titulación.\n Curso 16-17") %>%
  hc_xAxis(categories = por_titulacion_ano$Titulacion) %>%
  hc_add_series(name = "% Alumnos", data = list(
    list( y = por_titulacion_ano$Total16_17[1],
          name = por_titulacion_ano$Titulacion[1],dataLabels = list(enabled=TRUE)),
    list( y = por_titulacion_ano$Total16_17[2],
          name = por_titulacion_ano$Titulacion[2],dataLabels = list(enabled=TRUE)),
    list( y = por_titulacion_ano$Total16_17[3],
          name = por_titulacion_ano$Titulacion[3], sliced = TRUE),
    list( y = por_titulacion_ano$Total16_17[4],
```

```
        list( y = por_titulacion_ano$Total16_17[5],
              name = por_titulacion_ano$Titulacion[5],dataLabels = list(enabled=TRUE))))
```

- Cuadros de Tablas: "Porcentaje que representa cada Titulación por año" y "Total matriculados por Titulación y año".

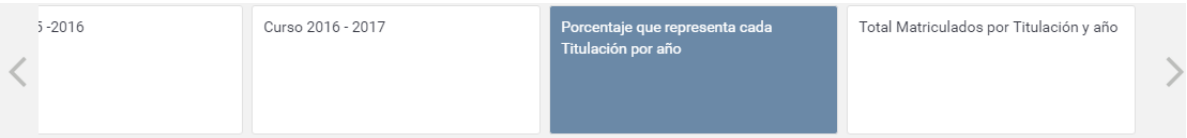

Figura 4.9: Storybard: Gráficos y Tablas

Las tablas están generadas a través de la librería **DT**, con diferentes argumentos de presentación, diseño y opciones varias como podemos ver en la que se refiere al *Porcentaje que representa cada Titulación por año* en la figura 4.10 donde, por ejemplo, se puede exportar la tabla a varios formatos como CSV o PDF. De forma similar es la que se refiere al *Total matriculados por Titulación y año*. Veamos el código utilizado para generar la tabla que vemos en la figura 4.10.

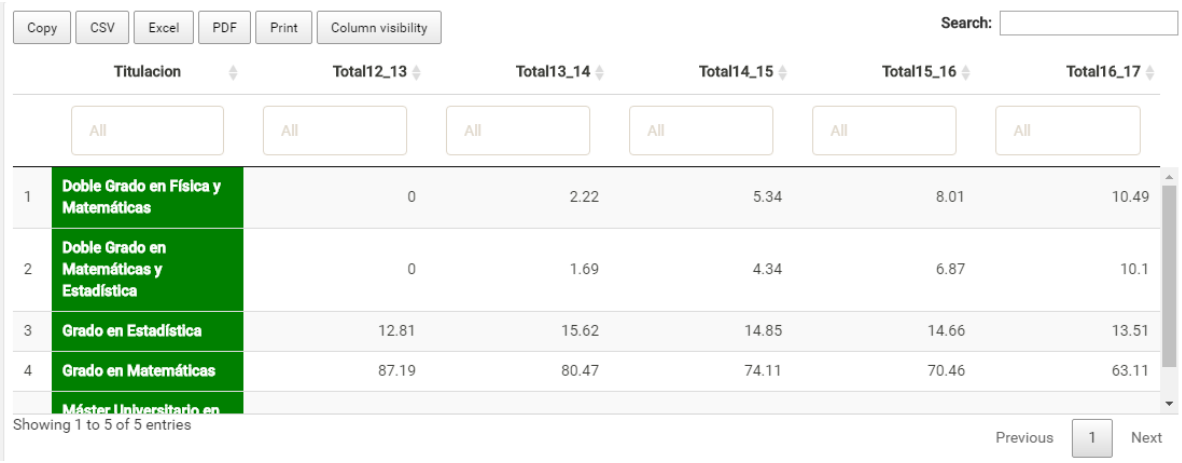

Figura 4.10: Storybard: Gráficos y Tablas

De forma similar es la que se refiere al *Total matriculados por Titulación y año*. Veamos el código utilizado para generar la tabla que vemos en la figura 4.10.

```
por_titulacion_ano %>%
  datatable(filter="top",   extensions = list("ColReorder" = NULL,
                      "Buttons" = NULL)
                      , options = list(
    dom = 'Bfrtip',
    buttons = list(list(extend='csv', filename = 'Informacion Tabla'),
                   list(extend='excel',filename = 'Informacion Tabla'),
                   list(extend='pdf',filename= 'Informacion Tabla'),
                   'copy','print', I('colvis')),
    colReorder = TRUE))  %>%
  formatStyle('Titulacion',  color = 'white', backgroundColor = 'green',
              fontWeight = 'bold')
```

CURSO Y TITULACIÓN

Esta página presenta una estructura de storyboard, similar a la de *Titulación*. En este caso, presentamos información relativa al porcentaje de alumnos que representa cada curso de una titulación. Tenemos esta información para cada titulación y año académico. Debido a que el master solamente consiste en un año, consideramos unicamente las titulaciones de grado y doble grado.

	Titulacion	Curso	Total12_13	Total13_14	Total14_15	Total15_16	Total16_17	Porcen12_13	Porcen13_14	Porcen14_15	Porcen15_16
1	Doble Grado en Física y Matemáticas	1	0	105	144	142	194	0	100	50.53	30.47
2	Doble Grado en Física y Matemáticas	2	0	0	141	199	189	0	0	49.47	42.7
3	Doble Grado en Física y Matemáticas	3	0	0	0	125	168	0	0	0	26.82

Figura 4.11: Página: Curso y Titulación

Encontramos en esta pagína:

- Tabla donde encontramos el número de total y el porcentaje de alumnos para cada titulación y curso para cada año académico, como vemos en la figura 4.11.
- Gráfico de sectores que muestran la información recogida en la tabla anterior. Por ejemplo, para el curso académico 2014 - 2015, se muestra lo siguiente:

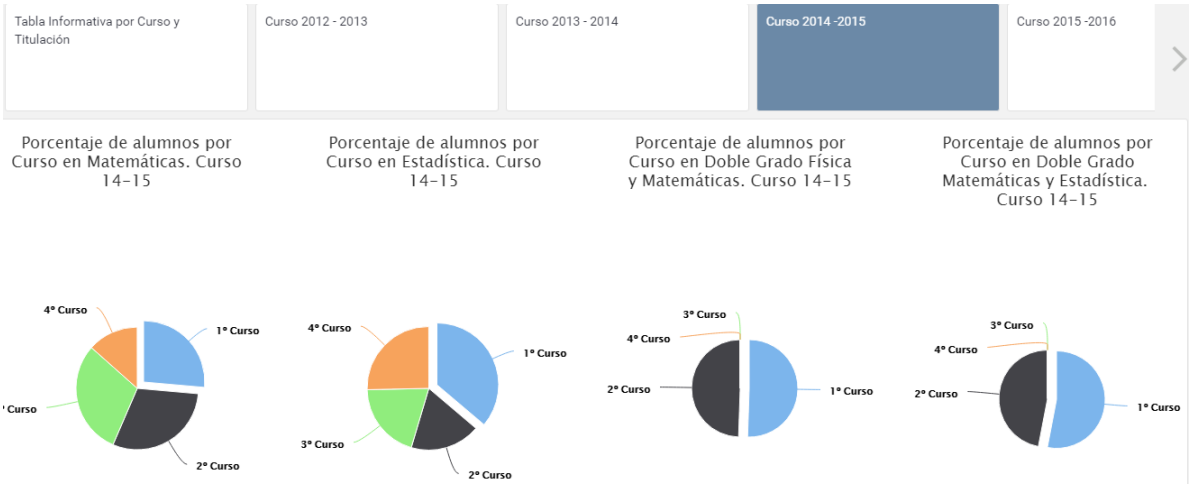

Figura 4.12: Porcentaje por curso de cada titulación

Como podemos ver los sectores se han construido a través de la librería **highchart** como en el caso anterior. Además, para tener una visualización de los 4 gráficos a la vez utilizamos la función *bscols* de la librería **crosstalk**, que permite poner elementos html uno junto al otro.

4.4.3.3. SERIES

Dentro de la pestaña **series** encontramos cuatro páginas: *Índices Informativos*, *Con Plotly*, *Con Dygraph* y *Comparación con Dygraph y Shiny*.

ÍNDICES INFORMATIVOS

En esta página encontramos 4 cajas de valor que van modificando lo que muestran según su valor.

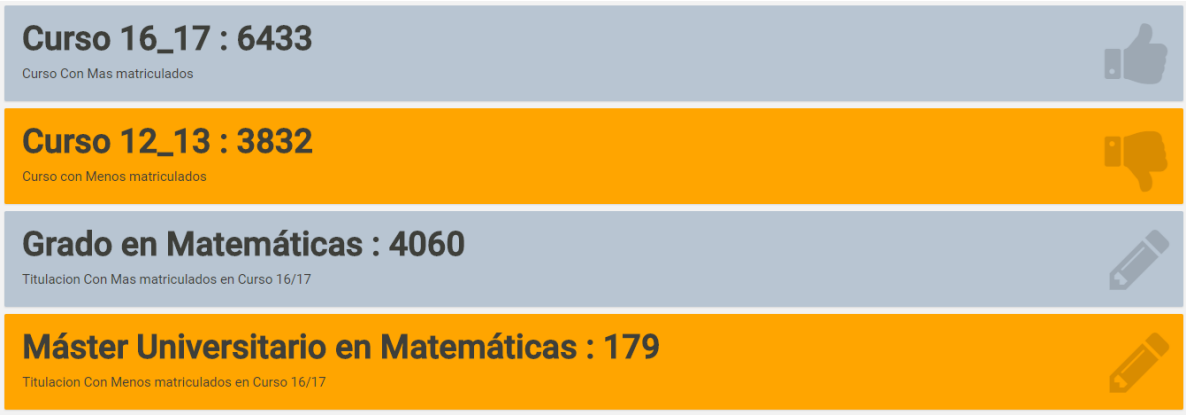

Figura 4.13: Indices informátivos

Vemos los siguiente información:

- Curso o año académico que tiene o ha tenido más matriculados. Se muestra el curso y el valor, es decir, el número de matriculados. Presenta como icono un pulgar hacia arriba. En este caso, el curso con más matriculados es 2016 - 2017.
- Curso o año académico que tiene o ha tenido menos matriculados. Se muestra el curso y el valor, es decir, el número de matriculados. Presenta como icono un pulgar hacia abajo y un fondo de color naranja. En este caso, el curso con menos matriculados es 2012 - 2013.
- Titulación, del curso 2016 - 2017, que presenta más matriculados. Se muestra la titulación y el valor. Presenta como icono un lápiz. En este caso la titulación con más alumnos es el Grado en Matemáticas.
- Titulación, del curso 2016 - 2017, que presenta menos matriculados. Se muestra la titulación y el valor. Presenta como icono un lápiz y un fondo de color naranja. En este caso la titulación con menos alumnos es el Master en Matemáticas. Veamos el código de este caso para ver un ejemplo de como se construyen este tipo de widgets.

```
aux = total_matriculados_ano_titulacion$Titulacion[
  which.min(total_matriculados_ano_titulacion$Total16_17)]

aux2 = min(total_matriculados_ano_titulacion$Total16_17)
valueBox(paste(as.character(aux),":",aux2), icon = "fa-pencil",
         col = "orange")
```

CON PLOTLY

En esta página podemos ver la evolución del número de matriculados para cada titulación a través de los sucesivos cursos académicos.

Lo interesante de este apartado es ejemplificar un uso de la librería **plotly**. En este caso, lo que se ha hecho es construir el gráfico a través de comandos de la librería **ggplot2** y, después, pasarle la función *ggplotly* para pasarlo a un objeto *plotly*. El código sería el siguiente:

```
p = total_matriculados_ano_titulacion %>%
  gather(Curso, Matriculados,  -Titulacion) %>%
  mutate(Curso = substr(Curso,6,10 )) %>%
  arrange(Titulacion) %>%
  ggplot(aes(x = Curso, y = Matriculados, group = Titulacion, color = Titulacion)) +
  geom_line() +
  theme(legend.position="none")

 ggplotly(p)
```

Con la función *gather* de la librería **tidyr** reagrupamos las columnas en dos variables:

- Curso, que toma los valores: Total12_13, Total13_14,. . . ,Total16_17
- Matriculados, toma los valores que estaban distribuidos por las 6 columnas anteriores, según la titulación y el Curso.

De esta forma transformamos los datos pasando de la figura 4.14 a la 4.15 (recorte de la tabla original)

	Titulacion	Total12_13	Total13_14	Total14_15	Total15_16	Total16_17
	<fctr>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>
1	Doble Grado en Física y Matemáticas	0	105	285	466	675
2	Doble Grado en Matemáticas y Estadística	0	80	232	400	650
3	Grado en Estadística	491	738	793	853	869
4	Grado en Matemáticas	3341	3802	3958	4100	4060
5	Máster Universitario en Matemáticas	0	0	73	0	179

Figura 4.14: Tabla inicial

A tibble: 25 × 3

Titulacion	Curso	Matriculados
<fctr>	<chr>	<dbl>
Doble Grado en Física y Matemáticas	Total12_13	0
Doble Grado en Matemáticas y Estadística	Total12_13	0
Grado en Estadística	Total12_13	491
Grado en Matemáticas	Total12_13	3341
Máster Universitario en Matemáticas	Total12_13	0
Doble Grado en Física y Matemáticas	Total13_14	105
Doble Grado en Matemáticas y Estadística	Total13_14	80

Figura 4.15: Tabla intermedia tras aplicar la función gather

Con la función *mutate*, del paquete **dplyr**, modificamos la variable Curso quedandonos solo con el dato númerico que determina el Curso académico, como podemos ver en la figura 4.16. Con la función *arrange*, de la librería **dplyr**, ordenamos de manera descendente según la titulación, es decir, en este caso en orden alfabético.

A tibble: 25 × 3

Titulacion	Curso	Matriculados
<fctr>	<chr>	<dbl>
Doble Grado en Física y Matemáticas	12_13	0
Doble Grado en Matemáticas y Estadística	12_13	0
Grado en Estadística	12_13	491
Grado en Matemáticas	12_13	3341

Figura 4.16: Tabla intermedia tras aplicar la función mutate

Una vez preparado los datos para realizar el gráfico que queremos utilizamos la función *ggplot* y *ggplotly* obteniendo un gráfico interactivo, como el que podemos ver en la figura 4.17, donde al pasar por algún punto nos muestra los datos referente a la titulación, valor y curso, permite hacer zoom, filtrado de zonas, etc.

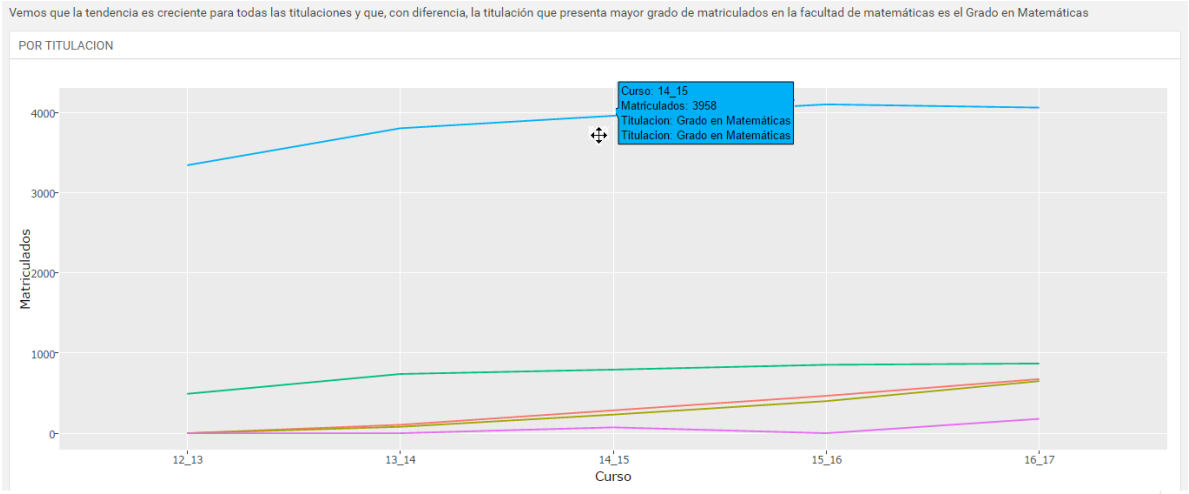

Figura 4.17: Gráfico con Plotly

CON DYGRPAH

En este apartado vamos a mostrar la misma información que visualizamos en la página anterior pero a partir de generar las series temporales con la librería **dygraph**. De esta forma, tenemos asginado cada curso académico a enero del año, es decir: Curso 2012-2013 corresponde a Enero de 2013.

Con *dygrpah* obtenemos un gráfico interactivo donde para cada punto temporal nos muestra información relativa a:

- Curso: Muestra el año, es decir Curso 2012-2013 mostrará 2013.
- DG F Y M: Número de matriculados en el Dobre Grado de Física y Matemáticas.
- DG M Y E: Número de matriculados en el Dobre Grado de Matemáticas y Estadística.
- E: Número de matriculados en el Grado de Estadística.
- M: Número de matriculados en el Grado de Matemáticas.
- M M: Número de matriculados en el Master en Matemáticas.

Obtenemos un gráfico interactivo, como el que vemos en la figura 4.18, donde podemos filtrar según la temporalidad (curso) deseado.

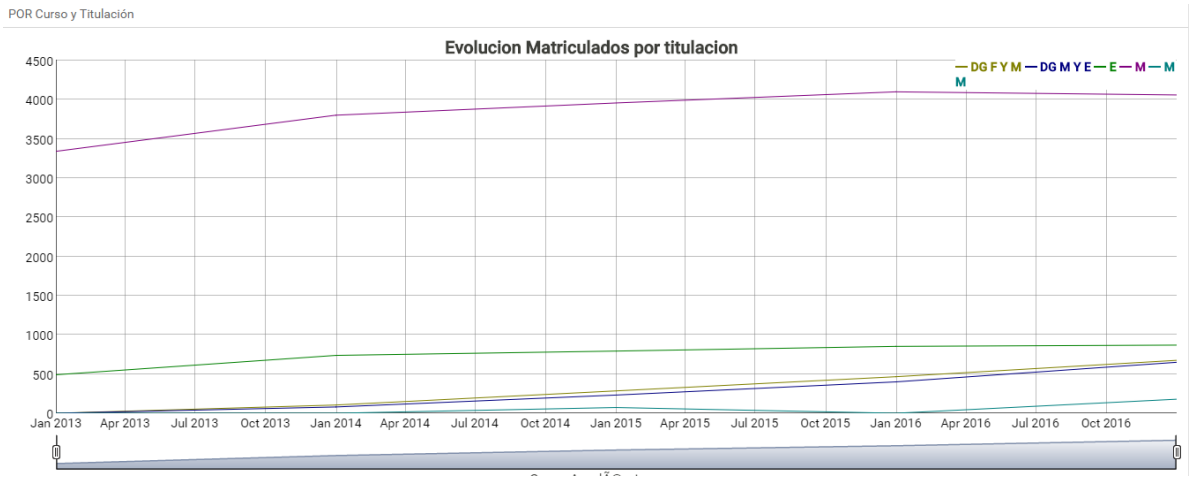

Figura 4.18: Gráfico con Dygraph

COMPARACIÓN CON DYGRAPH Y SHINY

En esta pagína introducimos, dentro de nuestro cuadro de mandos, contruido mendiante la libreria **flexdashboard**, una aplicación **shiny** interactiva con un gráfico generado con **dygraph**.

La aplicación consiste en un gráfico que muestra la evolución del número de matriculados de dos títulaciones seleccionadas por el usuario. Se compara el número de matriculados de esas dos titulaciones dibujandolas cada una en un eje , tal como podemos ver en la figura 4.19, donde comparamos la titulación en Matemáticas y Estadística.

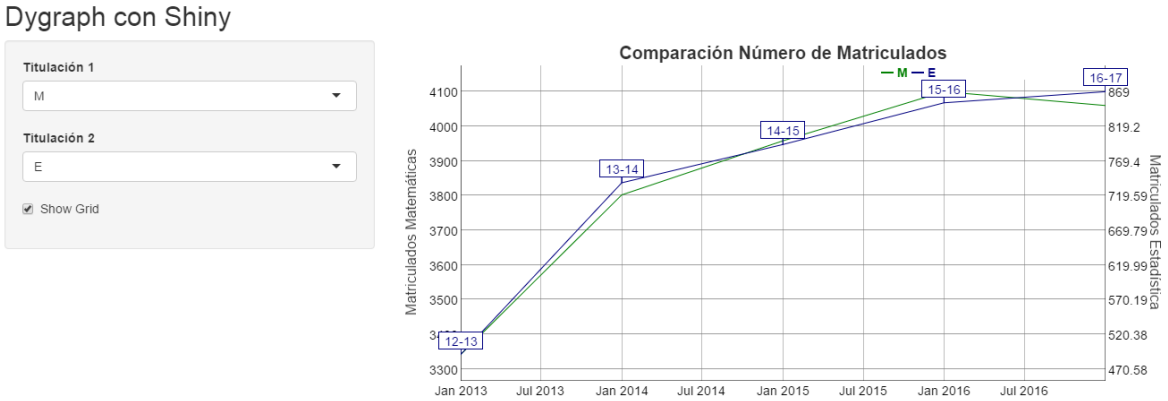

Figura 4.19: Shiny y Dygraph

De esta forma, la aplicación permite seleccionar:

- Titulación Principal a comparar.
- Titulación Secundaria a comparar.
- Mostrar rejilla o no.

Además mostramos unas etiquetas que hacen referencia al curso académíco, ej: 12-13, donde al pasar el cursor por encima da una mayor información, por ejemplo en el caso anterior mostraría “Curso 2012-2013”, como podemos ver en la figura 4.20. Esto la hacemos con la función *dyAnnotation* de la librería **dygraph**.

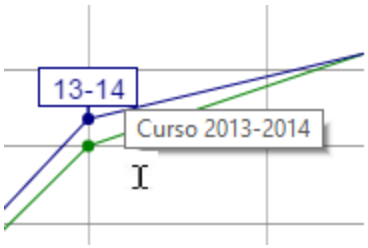

Figura 4.20: Anotaciones con Dygraph

Para incorporar a nuestro cuadro de mandos contruido con flexdashboard una aplicación shiny, que hemos creado previamente, utilizamos la función *shinyAppFile* donde sólo tenemos que especificar la ruta donde se encuentra la app shiny[1].

4.4.3.4. MATRICULADOS

Dentro de la pestaña **matriculados** encontramos dos páginas: *"Comparación Total Matriculados"* y *"Matriculados y Repetidores"*.

COMPARACIÓN TOTAL MATRICULADOS

En esta pagína podemos encontrar una división por pestañas dentro de la misma que corresponde a:

1. Comparación de alumnos matriculados en los cursos académicos 2016-2017 y 2015-2016 en las diferentes asignaturas de cada titulación.
2. Comparación de alumnos matriculados en los cursos académicos 2016-2017 y 2014-2015 en las diferentes asignaturas de cada titulación.

La división de pestañas, que podemos ver la figura 4.21, la obtenemos utilizando la especificación *tabset* como podemos ver en el siguiente código:

```
Column {.tabset}
-------------------------------------
```

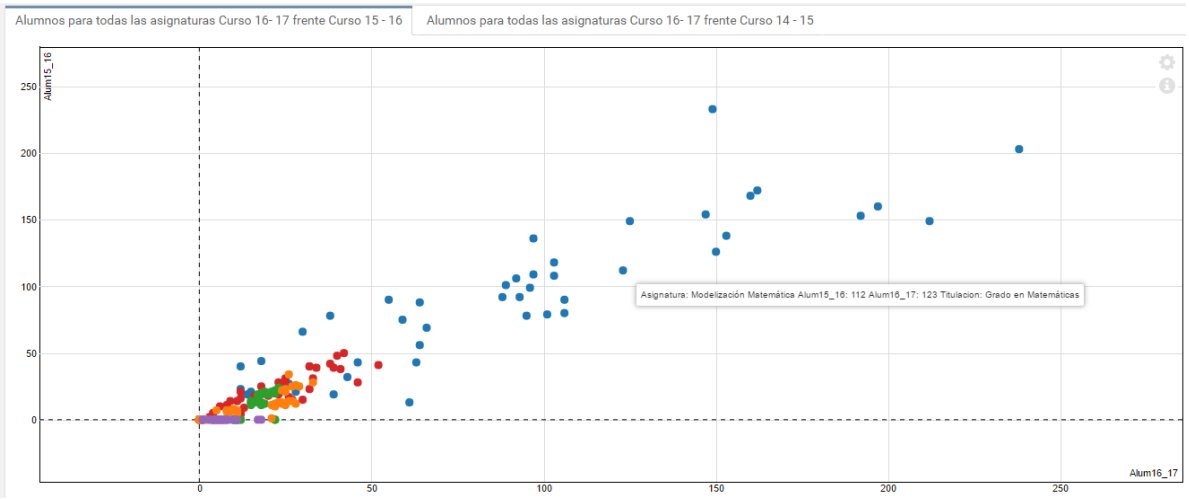

Figura 4.21: Gráficos con scatterD3 y división de pestañas en página.

[1]código: Anexo, APLICACIÓN: "Dygraph con Shiny"

Para la creación de estos gráficos, utilizamos el paquete **scatterD3** que nos permite:

- Matriculados para cada asigantura en ambos cursos.
- Diferenciación y filtrado según titulación.
- Movimiento del gráfico.
- Información general del gráfico, como vemos en la figura 4.22, que obtenemos al pulsar sobre el botón de información

Figura 4.22: Información sobre el gráfico

MATRICULADOS Y REPETIDORES

En este apartado, vamos a tener una visualización del número de repetidores de las asignaturas por titulación frente al número de matriculados total en el año académico 2016- 2017 de cada titulación.

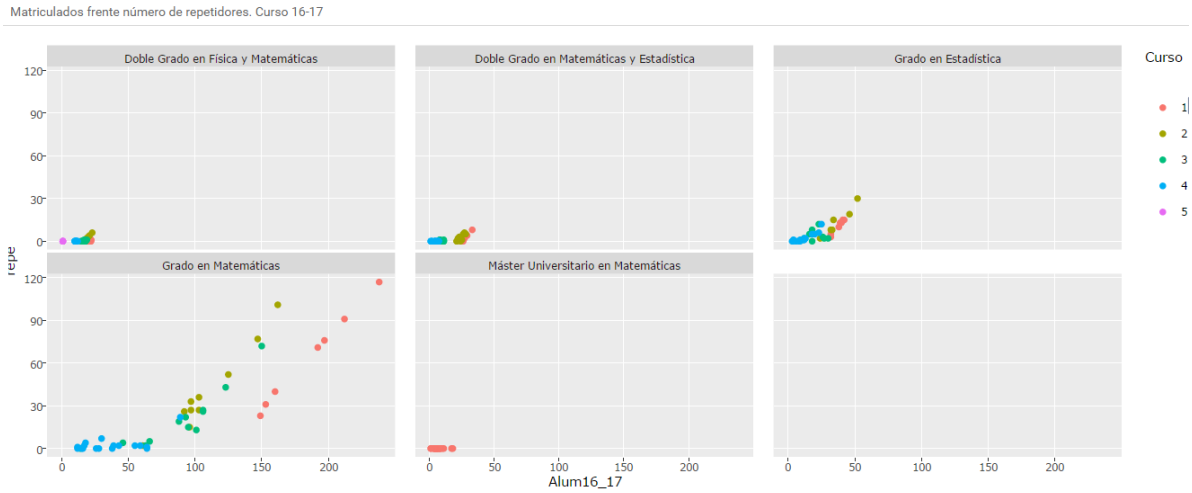

Figura 4.23: Matriculados frente a repetidores 2016-2017

Esta visulaización la hemos realizado con **ggplot** y **gplotly**, de manera que podems otener una visiualización a la vez de todas las titulaciones, como podemos ver en la figura 4.23 de manera interactiva. Al pasar el cursor por alguno de los puntos nos da información sobre a que títulación pertenece, el curso, la asignatura y el valor del número de matriculados como el de repetidores, cómo podemos ver en la figura 4.24, donde se muestra la información referente a una asignatura de Estadística.

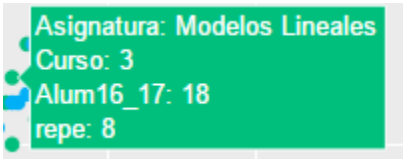

Figura 4.24: Información mostrada en Matriculados frente a repetidores 2016-2017

Nos permite:

- Poder hacer zoom dentro las diferentes titulaciones.
- Seleccionar los cursos que queremos que se muestren.

Para obtener la visualización de los gráficos de todas las titulaciones a la vez utilizamos la función de *facet_wrap(~Titulacion)* de **ggplot2**

4.4.3.5. INDICADORES DE REPETIDORES

Dentro de la pestaña **matriculados** encontramos dos páginas: *"Repetidores Matemáticas"* y *"Repetidores Estadística"*.

En ambas páginas encontramos unos indicadores sobre el número de repetidores, ya sea para Matemáticas o Estadística, por cada curso.

Los indicadores son los denominados "velocímetros" que muestran el porcentaje medio de alumnos que son repetidores del total de matriculados para cada curso de Matemáticas o Estadística.

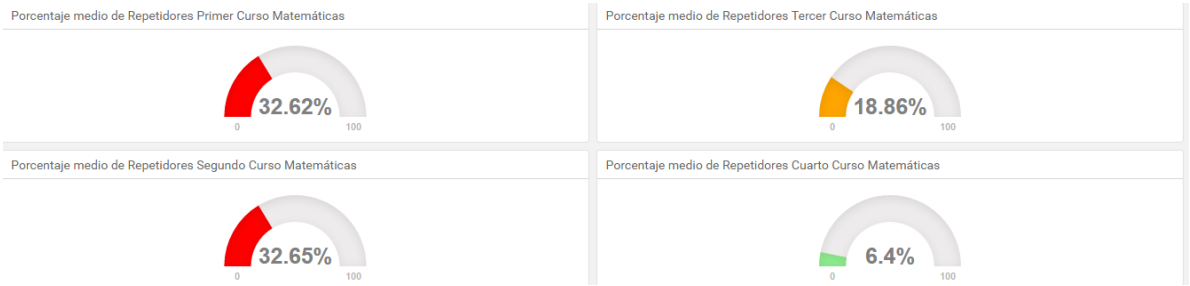

Figura 4.25: Indicadores repetidores en Grado de Matemáticas

El valor de los porcentajes se muestran en una barra que muestran su importancia según lo "llena" que esté y el color que tenga, como podemos observar en la figura 4.25. Los colores posibles son verde(casos más favorables, donde el porcentaje de repetidores es muy bajo), naranja (porcentaje considerable de repetidores) y rojo (porcentaje alto.). Tal como lo hemos codificado los colores los tomará según los rango de valores siguientes:

- Verde: 0 % - 10 %
- Naranja: 11 % - 20 %
- Rojo: 21 % - 100 %

Para obtener estos indicadores utilizamos las funciones *gauges* y *rate* del paquete **flexdashboard** como vemos en el siguiente código, donde encontramos un ejemplo de un gauge:

```
rate = as.numeric(porcen_medio_repe %>%
  filter(Titulacion == "Grado en Matemáticas", Curso == 2) %>%
  ungroup() %>%
  select(porcentaje_medio_repe))

rate = round(rate, 2)

gauge(rate, min = 0, max = 100, symbol = '%',
      gaugeSectors(colors = c("lightgreen", "orange", "red"), success = c(0, 10),
                   warning = c(11, 20), danger = c(21, 100)),
      abbreviateDecimals = 2, abbreviate = TRUE)
```

4.4.3.6. ALUMNOS POR ASIGNATURA

Dentro de la pestaña **alumnos por asignatura** encontramos tres páginas: *"Matemáticas"*, *"Estadística"* y *"Con Shiny"*.

La estructura de las páginas *"Matemáticas"* y *"Estadística"* son análogas. Se presenta una visualización con formato de *storyboard* diviendo por cursos. Para cada curso mostramos el número de alumnos de las asignaturas del curso 2016-2017 frente al curso 2015-2016.

Para ello utilizamos el paquete **highchart** para mostrar gráficos de barra con el argumento *hc_chart(type = "bar")*.

Podemos obtener diferente tipo de visualizaciones de gráficos de barra según las opciones que determinemos como por ejemplo:

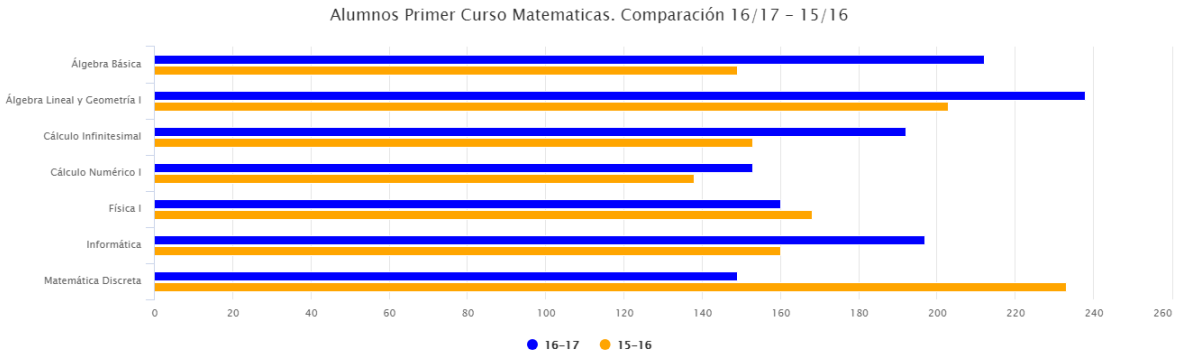

Figura 4.26: Opción 1 highchart bar

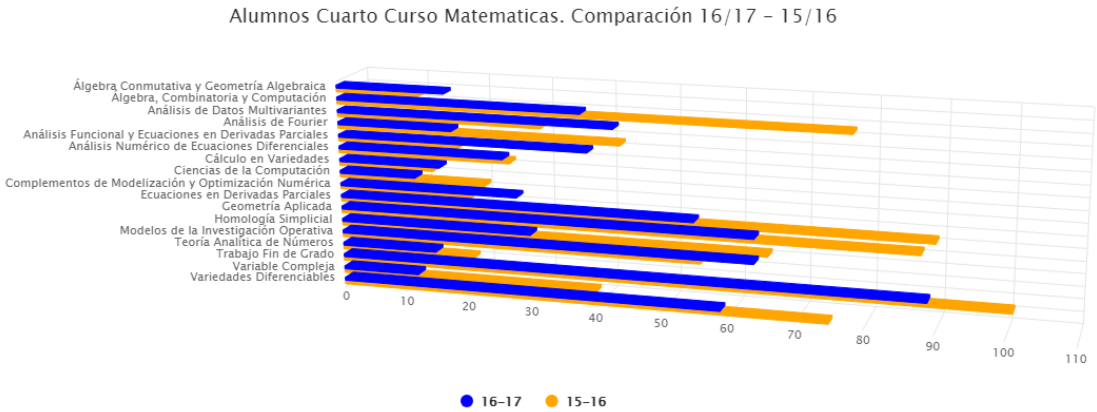

Figura 4.27: Opción 2 highchart bar

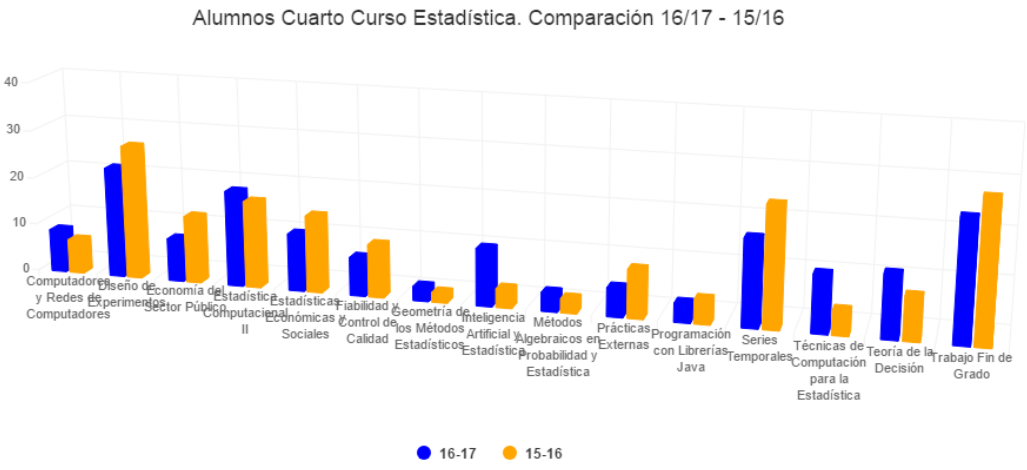

Figura 4.28: Opción 3 highchart bar

CON SHINY

En este apartado mostramos un aplicación que auna las diferentes opciones de las dos páginas anteriores en una única aplicación **shiny** [2], donde podemos seleccionar:

- Titulación.
- Curso.
- Años académicos a comparar.

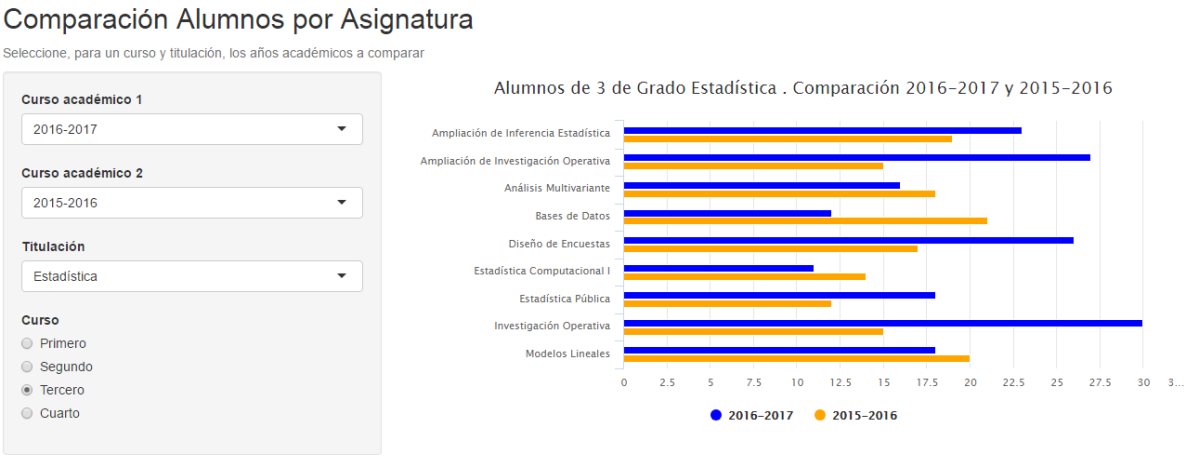

Figura 4.29: Alumnos por asignatura con Shiny

[2] código: Anexo .APLICACIÓN: Comparación Alumnos por Asignatura"

4.5. ANEXO CÓDIGO COMPLETO

```
library(scatterD3)
library(shiny)
library(flexdashboard)
library(tidyverse)
library(gridExtra)
library(readxl)
library(DT)
library(plotly)
library(xts)
library(dygraphs)
library(zoo)
library(leaflet)
library(crosstalk)
library(d3scatter)
library(highcharter)

datos_GM = readxl::read_excel("estadmatr_Matematicas.xlsx")
datos_GE = read_excel("estadmatr_Estadistica.xlsx")
datos_MUM = read_excel("estadmatr_Matematicas_MUM.xlsx")
datos_DGFM = read_excel("estadmatr_Fisica_Matematicas.xlsx")
datos_DGME = read_excel("estadmatr_Matematicas_Estadistica.xlsx")

datos_todos = rbind(datos_GM,datos_GE,datos_DGFM,datos_DGME,datos_MUM)
colnames(datos_todos)[c(10,11,12,13,14,18,19,20,23)] =
  c("Alum12_13",   "Alum13_14", "Alum14_15", "Alum15_16",   "Alum16_17","Grupos16_17","Activ","Media_po


datos_todos_r = datos_todos %>%
  select(Centro,Titulación,Asignatura,curso,Alum12_13,
         Alum13_14,Alum14_15,Alum15_16,Alum16_17,repe,
         tantorepe,Grupos16_17,Activ,DptoPral)

datos_todos_r$Centro = factor(datos_todos_r$Centro)
datos_todos_r$Titulación = factor(datos_todos_r$Titulación)
datos_todos_r$Asignatura = factor(datos_todos_r$Asignatura)
datos_todos_r$DptoPral = factor(datos_todos_r$DptoPral)
datos_todos_r$Activ = factor(datos_todos_r$Activ)
datos_todos_r$curso = factor(datos_todos_r$curso)
datos_todos_r$Grupos16_17 = factor(datos_todos_r$Grupos16_17)

datos = datos_todos_r

colnames(datos)= c("Centro", "Titulacion", "Asignatura", "Curso",
                   "Alum12_13", "Alum13_14",  "Alum14_15"
                           , "Alum15_16",   "Alum16_17",
                   "repe", "tantorepe", "Grupos16_17", "Activ", "DptoPral")



total_matriculados_ano = datos %>%
  select(Alum12_13:Alum16_17) %>%
```

```
  summarise(Total12_13 = sum(Alum12_13), Total13_14 = sum(Alum13_14),
            Total14_15 = sum(Alum14_15),
            Total15_16 = sum(Alum15_16), Total16_17 = sum(Alum16_17, na.rm = T))

 #Para Doble grado hay asignaturas desiertas: NA

total_matriculados_ano_titulacion =  datos %>%
  select(Titulacion, Alum12_13:Alum16_17) %>%
  group_by(Titulacion) %>%
  summarise(Total12_13 = sum(Alum12_13), Total13_14 = sum(Alum13_14),
            Total14_15 = sum(Alum14_15),
            Total15_16 = sum(Alum15_16), Total16_17 = sum(Alum16_17, na.rm = T))


## Porcentajes de alumno de cada titulacion por año

por_titulacion_ano = total_matriculados_ano_titulacion

for( i in 1:5 ){
por_titulacion_ano[i,2:6] = round(total_matriculados_ano_titulacion[i,-1] /
                                  total_matriculados_ano * 100, 2)
}


##### GRAFICOS PORCENTAJE POR TITULACION
grafico_sector_titulacion = function(Totalano){

daux = por_titulacion_ano[Totalano]
daux = cbind(por_titulacion_ano[,1], daux)
colnames(daux)[2]  = "y"

 ggplot(daux, aes(x="", y= y , fill=Titulacion))+
  geom_bar(width = 1, stat = "identity") +
   coord_polar("y", start=0) +
   theme(axis.text.x=element_blank()) +
   ggtitle(paste("Porcentaje de alumnos por Titulación.\n Curso",
                 substr(Totalano, 6,10))) +
   ylab("")+
   xlab("")
}


leaflet() %>%
  addTiles() %>%
  addMarkers(lng=-5.98814, lat=37.35945,
             popup="Facultad de Matemáticas")%>%
  setView(lng=-5.98814, lat=37.35945, zoom = 18)
#grafico_sector_titulacion("Total12_13")

highchart() %>%
  hc_chart(type = "pie") %>%
  hc_title(text ="Porcentaje de alumnos por Titulación.\n Curso 12-13") %>%
  hc_xAxis(categories = por_titulacion_ano$Titulacion) %>%
```

```
  hc_add_series(name = "% Alumnos", data =
                  list(list( y = por_titulacion_ano$Total12_13[1],
                             name = por_titulacion_ano$Titulacion[1],
                             dataLabels = list(enabled=FALSE)),
                       list( y = por_titulacion_ano$Total12_13[2],
                             name = por_titulacion_ano$Titulacion[2],
                             dataLabels = list(enabled=FALSE)),
                       list( y = por_titulacion_ano$Total12_13[3],
                             name = por_titulacion_ano$Titulacion[3], sliced = TRUE),
                       list( y = por_titulacion_ano$Total12_13[4],
                             name = por_titulacion_ano$Titulacion[4]),
                       list( y = por_titulacion_ano$Total12_13[5],
                             name = por_titulacion_ano$Titulacion[5],
                             dataLabels = list(enabled=FALSE))))

#grafico_sector_titulacion("Total13_14")

highchart() %>%
  hc_chart(type = "pie") %>%
  hc_title(text ="Porcentaje de alumnos por Titulación.\n Curso 13-14") %>%
  hc_xAxis(categories = por_titulacion_ano$Titulacion) %>%
  hc_add_series(name = "% Alumnos", data =
                  list(list( y = por_titulacion_ano$Total13_14[1],
                             name = por_titulacion_ano$Titulacion[1],
                             dataLabels = list(enabled=TRUE)),
                       list( y = por_titulacion_ano$Total13_14[2],
                             name = por_titulacion_ano$Titulacion[2],
                             dataLabels = list(enabled=TRUE)),
                       list( y = por_titulacion_ano$Total13_14[3],
                             name = por_titulacion_ano$Titulacion[3],
                             sliced = TRUE),
                       list( y = por_titulacion_ano$Total13_14[4],
                             name = por_titulacion_ano$Titulacion[4]),
                       list( y = por_titulacion_ano$Total13_14[5],
                             name = por_titulacion_ano$Titulacion[5],
                             dataLabels = list(enabled=FALSE))))
#grafico_sector_titulacion("Total14_15")

highchart() %>%
  hc_chart(type = "pie") %>%
  hc_title(text ="Porcentaje de alumnos por Titulación.\n Curso 14-15") %>%
  hc_xAxis(categories = por_titulacion_ano$Titulacion) %>%
  hc_add_series(name = "% Alumnos", data =
                  list(list( y = por_titulacion_ano$Total14_15[1],
                             name = por_titulacion_ano$Titulacion[1],
                             dataLabels = list(enabled=TRUE)),
                       list( y = por_titulacion_ano$Total14_15[2],
                             name = por_titulacion_ano$Titulacion[2],
                             dataLabels = list(enabled=TRUE)),
                       list( y = por_titulacion_ano$Total14_15[3],
                             name = por_titulacion_ano$Titulacion[3],
                             sliced = TRUE),
                       list( y = por_titulacion_ano$Total14_15[4],
```

```
                              name = por_titulacion_ano$Titulacion[4]),
                        list( y = por_titulacion_ano$Total14_15[5],
                              name = por_titulacion_ano$Titulacion[5],
                              dataLabels = list(enabled=TRUE))))
#grafico_sector_titulacion("Total15_16")

highchart() %>%
  hc_chart(type = "pie") %>%
  hc_title(text ="Porcentaje de alumnos por Titulación.\n Curso 15-16") %>%
  hc_xAxis(categories = por_titulacion_ano$Titulacion) %>%
  hc_add_series(name = "% Alumnos", data =
                  list(list( y = por_titulacion_ano$Total15_16[1],
                              name = por_titulacion_ano$Titulacion[1],
                              dataLabels = list(enabled=TRUE)),
                        list( y = por_titulacion_ano$Total15_16[2],
                              name = por_titulacion_ano$Titulacion[2],
                              dataLabels = list(enabled=TRUE)),
                        list( y = por_titulacion_ano$Total15_16[3],
                              name = por_titulacion_ano$Titulacion[3],
                              sliced = TRUE),
                        list( y = por_titulacion_ano$Total15_16[4],
                              name = por_titulacion_ano$Titulacion[4]),
                        list( y = por_titulacion_ano$Total15_16[5],
                              name = por_titulacion_ano$Titulacion[5],
                              dataLabels = list(enabled=FALSE))))
#grafico_sector_titulacion("Total16_17")

highchart() %>%
  hc_chart(type = "pie") %>%
  hc_title(text ="Porcentaje de alumnos por Titulación.\n Curso 16-17") %>%
  hc_xAxis(categories = por_titulacion_ano$Titulacion) %>%
  hc_add_series(name = "% Alumnos", data =
                  list(list( y = por_titulacion_ano$Total16_17[1],
                              name = por_titulacion_ano$Titulacion[1],
                              dataLabels = list(enabled=TRUE)),
                        list( y = por_titulacion_ano$Total16_17[2],
                              name = por_titulacion_ano$Titulacion[2],
                              dataLabels = list(enabled=TRUE)),
                        list( y = por_titulacion_ano$Total16_17[3],
                              name = por_titulacion_ano$Titulacion[3],
                              sliced = TRUE),
                        list( y = por_titulacion_ano$Total16_17[4],
                              name = por_titulacion_ano$Titulacion[4]),
                        list( y = por_titulacion_ano$Total16_17[5],
                              name = por_titulacion_ano$Titulacion[5],
                              dataLabels = list(enabled=TRUE))))
por_titulacion_ano %>%
  datatable(filter="top",   extensions = list("ColReorder" = NULL,
                      "Buttons" = NULL)
                      , options = list(
    dom = 'Bfrtip',
    buttons = list(list(extend='csv', filename = 'Informacion Tabla'),
                   list(extend='excel',filename = 'Informacion Tabla'),
```

```
                    list(extend='pdf', filename= 'Informacion Tabla'),
                    'copy','print', I('colvis')),
    colReorder = TRUE))  %>%
  formatStyle('Titulacion',  color = 'white',
              backgroundColor = 'green', fontWeight = 'bold')

total_matriculados_ano_titulacion %>%
  datatable(filter="top",  extensions = list("ColReorder" = NULL,
                        "Buttons" = NULL)
                        , options = list(
    dom = 'Bfrtip',
    buttons = list(list(extend='csv', filename = 'Informacion Tabla'),
                    list(extend='excel', filename = 'Informacion Tabla'),
                    list(extend='pdf', filename= 'Informacion Tabla'),
                    'copy','print', I('colvis')),
    colReorder = TRUE))  %>%
  formatStyle('Titulacion',  color = 'white',
              backgroundColor = 'green', fontWeight = 'bold')


total_curso_titulacion = datos %>%
  group_by(Titulacion, Curso) %>%
  summarise(Total12_13 = sum(Alum12_13),
            Total13_14 = sum(Alum13_14), Total14_15 = sum(Alum14_15),
            Total15_16 = sum(Alum15_16), Total16_17 = sum(Alum16_17, na.rm = T))

por_curso_titulacion = total_curso_titulacion
por_curso_titulacion

str(total_curso_titulacion)
for (i in 1:5) {
  por_curso_titulacion[i,-(1:2)] =
    round(total_curso_titulacion[i,-(1:2)] /
            total_matriculados_ano_titulacion[1,-1] * 100,2)
}
for (i in 6:10) {
  por_curso_titulacion[i,-(1:2)] =
    round(total_curso_titulacion[i,-(1:2)] /
            total_matriculados_ano_titulacion[2,-1] * 100,2)
}
for (i in 11:14) {
  por_curso_titulacion[i,-(1:2)] =
    round(total_curso_titulacion[i,-(1:2)] /
            total_matriculados_ano_titulacion[3,-1] * 100,2)
}
for (i in 15:18) {
  por_curso_titulacion[i,-(1:2)] =
    round(total_curso_titulacion[i,-(1:2)] /
            total_matriculados_ano_titulacion[4,-1] * 100,2)
}
por_curso_titulacion[19,-(1:2)] =
  round(total_curso_titulacion[19,-(1:2)] /
          total_matriculados_ano_titulacion[5,-1] * 100,2)
```

```
por_curso_titulacion

#NA como ceros
por_curso_titulacion[is.na(por_curso_titulacion)] = 0
por_curso_titulacion




grafico_sector_curso_titulacion = function(Totalano, titulacion){

  daux = por_curso_titulacion %>%
    filter(Titulacion == titulacion)

  daux2 = daux[Totalano]
  daux = cbind(daux[,2], daux2)

  colnames(daux)[2]  = "y"

  ggplot(daux, aes(x="", y= y , fill=Curso))+
    geom_bar(width = 1, stat = "identity") +
    coord_polar("y", start=0) +
    theme(axis.text.x=element_blank()) +
    ggtitle(paste("Porcentaje de alumnos por Curso.\n",
                  titulacion,"\nCurso", substr(Totalano, 6,10))) +
    ylab("")+
    xlab("")
}




# PAra cada año muestre lso porcentajes en las 4 titulaciones ( No tiene sentido en master
# ya que solo es un curso)
grafico_sector_curso_ano = function(Totalano){

  p1 = grafico_sector_curso_titulacion(Totalano,
                                       "Grado en Matemáticas")
  p2 = grafico_sector_curso_titulacion(Totalano,
                                       "Grado en Estadística")
  p3 = grafico_sector_curso_titulacion(Totalano,
                                       "Doble Grado en Física y Matemáticas")
  p4 = grafico_sector_curso_titulacion(Totalano,
                                       "Doble Grado en Matemáticas y Estadística")

  grid.arrange(p1, p2, p3, p4, ncol=2)

}
aux = names(which.max(total_matriculados_ano))
aux2 = max(total_matriculados_ano)
valueBox(paste("Curso", substr(aux, 6,10),":", aux2),
         icon = "fa-thumbs-up")
aux = names(which.min(total_matriculados_ano))
```

```
aux2 = min(total_matriculados_ano)
valueBox(paste("Curso", substr(aux, 6,10),":", aux2),
         icon = "fa-thumbs-down",
         color = "orange")
aux = total_matriculados_ano_titulacion$Titulacion[
  which.max(total_matriculados_ano_titulacion$Total16_17)]
aux2 = max(total_matriculados_ano_titulacion$Total16_17)
valueBox(paste(as.character(aux),":",aux2), icon = "fa-pencil")
aux = total_matriculados_ano_titulacion$Titulacion[
  which.min(total_matriculados_ano_titulacion$Total16_17)]
aux2 = min(total_matriculados_ano_titulacion$Total16_17)
valueBox(paste(as.character(aux),":",aux2), icon = "fa-pencil",
         col = "orange")
aux = merge(total_curso_titulacion, por_curso_titulacion,
            by = c("Titulacion", "Curso"))
colnames(aux)[3:7] = substr(colnames(aux)[3:7],1,10)
colnames(aux)[8:12] = sub("Total", "Porcen",
                          substr(colnames(aux)[8:12],1,10) )

aux %>%
 datatable(options=list(pageLength = 5,initComplete = JS(
"function(settings, json) {",
"$(this.api().table().header()).css({'background-color': '#228B22', 'color': '#fff'});",
    "}")),filter="top")
#grafico_sector_curso_ano("Total12_13")
Mat = por_curso_titulacion %>%
  filter(Titulacion == "Grado en Matemáticas")
Est = por_curso_titulacion %>%
  filter(Titulacion == "Grado en Estadística")
DobleFM = por_curso_titulacion %>%
  filter(Titulacion == "Doble Grado en Física y Matemáticas")
DobleME = por_curso_titulacion %>%
  filter(Titulacion == "Doble Grado en Matemáticas y Estadística")

p1 = highchart() %>%
  hc_chart(type = "pie") %>%
  hc_title(text ="Porcentaje de alumnos por Curso en \n Matemáticas. Curso 12-13") %>%
  hc_xAxis(categories = Mat$Curso) %>%
  hc_add_series(name = "% Alumnos",
                data = list(list( y = Mat$Total12_13[1], name = "1º Curso",
                                  sliced = TRUE),
                            list( y = Mat$Total12_13[2], name = "2º Curso"),
                            list( y = Mat$Total12_13[3], name = "3º Curso"),
                            list( y = Mat$Total12_13[4], name = "4º Curso")))

p2 = highchart() %>%
  hc_chart(type = "pie") %>%
  hc_title(text ="Porcentaje de alumnos por Curso en \n Estadística. Curso 12-13") %>%
  hc_xAxis(categories = Est$Curso) %>%
  hc_add_series(name = "% Alumnos",
                data = list(list( y = Est$Total12_13[1], name = "1º Curso",
                                  sliced = TRUE),
                            list( y = Est$Total12_13[2], name = "2º Curso"),
```

```
                            list( y = Est$Total12_13[3], name = "3º Curso"),
                            list( y = Est$Total12_13[4], name = "4º Curso")))

p3 = highchart() %>%
  hc_chart(type = "pie") %>%
  hc_title(
text ="Porcentaje de alumnos por Curso en \n
Doble Grado Física y Matemáticas. \n Curso 12-13") %>%
  hc_xAxis(categories = DobleFM$Curso) %>%
  hc_add_series(name = "% Alumnos",
                data = list(list( y = DobleFM$Total12_13[1], name = "1º Curso",
                                  sliced = TRUE),
                            list( y = DobleFM$Total12_13[2], name = "2º Curso"),
                            list( y = DobleFM$Total12_13[3], name = "3º Curso"),
                            list( y = DobleFM$Total12_13[4], name = "4º Curso")))

p4 = highchart() %>%
  hc_chart(type = "pie") %>%
  hc_title(
text ="Porcentaje de alumnos por Curso en \n Doble
Grado Matemáticas y Estadística.\n Curso 12-13") %>%
  hc_xAxis(categories = DobleME$Curso) %>%
  hc_add_series(name = "% Alumnos",
                data = list(list( y = DobleME$Total12_13[1], name = "1º Curso",
                                  sliced = TRUE),
                            list( y = DobleME$Total12_13[2], name = "2º Curso"),
                            list( y = DobleME$Total12_13[3], name = "3º Curso"),
                            list( y = DobleME$Total12_13[4], name = "4º Curso")))

bscols(p1,p2,p3,p4)


#grafico_sector_curso_ano("Total13_14")
p1 = highchart() %>%
  hc_chart(type = "pie") %>%
  hc_title(text ="Porcentaje de alumnos por Curso en \n Matemáticas. Curso 13-14") %>%
  hc_xAxis(categories = Mat$Curso) %>%
  hc_add_series(name = "% Alumnos",
                data = list(list( y = Mat$Total13_14[1], name = "1º Curso",
                                  sliced = TRUE),
                            list( y = Mat$Total13_14[2], name = "2º Curso"),
                            list( y = Mat$Total13_14[3], name = "3º Curso"),
                            list( y = Mat$Total13_14[4], name = "4º Curso")))

p2 = highchart() %>%
  hc_chart(type = "pie") %>%
  hc_title(text ="Porcentaje de alumnos por Curso en \n Estadística. Curso 13-14") %>%
  hc_xAxis(categories = Est$Curso) %>%
  hc_add_series(name = "% Alumnos",
                data = list(list( y = Est$Total13_14[1], name = "1º Curso",
                                  sliced = TRUE),
                            list( y = Est$Total13_14[2], name = "2º Curso"),
                            list( y = Est$Total13_14[3], name = "3º Curso"),
```

```
                            list( y = Est$Total13_14[4], name = "4º Curso")))

p3 = highchart() %>%
  hc_chart(type = "pie") %>%
  hc_title(
text ="Porcentaje de alumnos por Curso en \n
Doble Grado Física y Matemáticas.\n Curso 13-14") %>%
  hc_xAxis(categories = DobleFM$Curso) %>%
  hc_add_series(name = "% Alumnos",
                data = list(list( y = DobleFM$Total13_14[1], name = "1º Curso",
                                  sliced = TRUE),
                            list( y = DobleFM$Total13_14[2], name = "2º Curso"),
                            list( y = DobleFM$Total13_14[3], name = "3º Curso"),
                            list( y = DobleFM$Total13_14[4], name = "4º Curso")))

p4 = highchart() %>%
  hc_chart(type = "pie") %>%
  hc_title(
text ="Porcentaje de alumnos por Curso en \n Doble
Grado Matemáticas y Estadística.\n Curso 13-14") %>%
  hc_xAxis(categories = DobleME$Curso) %>%
  hc_add_series(name = "% Alumnos",
                data = list(list( y = DobleME$Total13_14[1], name = "1º Curso",
                                  sliced = TRUE),
                            list( y = DobleME$Total13_14[2], name = "2º Curso"),
                            list( y = DobleME$Total13_14[3], name = "3º Curso"),
                            list( y = DobleME$Total13_14[4], name = "4º Curso")))

bscols(p1,p2,p3,p4)
#grafico_sector_curso_ano("Total14_15")
p1 = highchart() %>%
  hc_chart(type = "pie") %>%
  hc_title(text ="Porcentaje de alumnos por Curso en \n Matemáticas. Curso 14-15") %>%
  hc_xAxis(categories = Mat$Curso) %>%
  hc_add_series(name = "% Alumnos",
                data = list(list( y = Mat$Total14_15[1], name = "1º Curso",
                                  sliced = TRUE),
                            list( y = Mat$Total14_15[2], name = "2º Curso"),
                            list( y = Mat$Total14_15[3], name = "3º Curso"),
                            list( y = Mat$Total14_15[4], name = "4º Curso")))

p2 = highchart() %>%
  hc_chart(type = "pie") %>%
  hc_title(text ="Porcentaje de alumnos por Curso en \n Estadística. Curso 14-15") %>%
  hc_xAxis(categories = Est$Curso) %>%
  hc_add_series(name = "% Alumnos",
                data = list(list( y = Est$Total14_15[1], name = "1º Curso",
                                  sliced = TRUE),
                            list( y = Est$Total14_15[2], name = "2º Curso"),
                            list( y = Est$Total14_15[3], name = "3º Curso"),
                            list( y = Est$Total14_15[4], name = "4º Curso")))

p3 = highchart() %>%
```

```
  hc_chart(type = "pie") %>%
  hc_title(
text ="Porcentaje de alumnos por Curso en \n
Doble Grado Física y Matemáticas.\n Curso 14-15") %>%
  hc_xAxis(categories = DobleFM$Curso) %>%
  hc_add_series(name = "% Alumnos",
                data = list(list( y = DobleFM$Total14_15[1], name = "1º Curso",
                                  sliced = TRUE),
                            list( y = DobleFM$Total14_15[2], name = "2º Curso"),
                            list( y = DobleFM$Total14_15[3], name = "3º Curso"),
                            list( y = DobleFM$Total14_15[4], name = "4º Curso")))

p4 = highchart() %>%
  hc_chart(type = "pie") %>%
  hc_title(
text="Porcentaje de alumnos por Curso en \n Doble
Grado Matemáticas y Estadística.\n Curso 14-15") %>%
  hc_xAxis(categories = DobleME$Curso) %>%
  hc_add_series(name = "% Alumnos",
                data = list(list( y = DobleME$Total14_15[1], name = "1º Curso",
                                  sliced = TRUE),
                            list( y = DobleME$Total14_15[2], name = "2º Curso"),
                            list( y = DobleME$Total14_15[3], name = "3º Curso"),
                            list( y = DobleME$Total14_15[4], name = "4º Curso")))

bscols(p1,p2,p3,p4)
#grafico_sector_curso_ano("Total15_16")
p1 = highchart() %>%
  hc_chart(type = "pie") %>%
  hc_title(text ="Porcentaje de alumnos por Curso en \n Matemáticas. Curso 15-16") %>%
  hc_xAxis(categories = Mat$Curso) %>%
  hc_add_series(name = "% Alumnos",
                data = list(list( y = Mat$Total15_16[1], name = "1º Curso",
                                  sliced = TRUE),
                            list( y = Mat$Total15_16[2], name = "2º Curso"),
                            list( y = Mat$Total15_16[3], name = "3º Curso"),
                            list( y = Mat$Total15_16[4], name = "4º Curso")))

p2 = highchart() %>%
  hc_chart(type = "pie") %>%
  hc_title(text ="Porcentaje de alumnos por Curso en \n Estadística. Curso 15-16") %>%
  hc_xAxis(categories = Est$Curso) %>%
  hc_add_series(name = "% Alumnos",
                data = list(list( y = Est$Total15_16[1], name = "1º Curso",
                                  sliced = TRUE),
                            list( y = Est$Total15_16[2], name = "2º Curso"),
                            list( y = Est$Total15_16[3], name = "3º Curso"),
                            list( y = Est$Total15_16[4], name = "4º Curso")))

p3 = highchart() %>%
  hc_chart(type = "pie") %>%
  hc_title(
text ="Porcentaje de alumnos por Curso en \n Doble
```

```
Grado Física y Matemáticas.\n Curso 15-16") %>%
  hc_xAxis(categories = DobleFM$Curso) %>%
  hc_add_series(name = "% Alumnos",
                data = list(list( y = DobleFM$Total15_16[1], name = "1º Curso",
                                  sliced = TRUE),
                            list( y = DobleFM$Total15_16[2], name = "2º Curso"),
                            list( y = DobleFM$Total15_16[3], name = "3º Curso"),
                            list( y = DobleFM$Total15_16[4], name = "4º Curso")))

p4 = highchart() %>%
  hc_chart(type = "pie") %>%
  hc_title(
text="Porcentaje de alumnos por Curso en \n Doble
Grado Matemáticas y Estadística. \n Curso 15-16") %>%
  hc_xAxis(categories = DobleME$Curso) %>%
  hc_add_series(name = "% Alumnos",
                data = list(list( y = DobleME$Total15_16[1], name = "1º Curso",
                                  sliced = TRUE),
                            list( y = DobleME$Total15_16[2], name = "2º Curso"),
                            list( y = DobleME$Total15_16[3], name = "3º Curso"),
                            list( y = DobleME$Total15_16[4], name = "4º Curso")))

bscols(p1,p2,p3,p4)
#grafico_sector_curso_ano("Total16_17")
p1 = highchart() %>%
  hc_chart(type = "pie") %>%
  hc_title(text ="Porcentaje de alumnos por Curso en \n Matemáticas. Curso 16-17") %>%
  hc_xAxis(categories = Mat$Curso) %>%
  hc_add_series(name = "% Alumnos",
                data = list(list( y = Mat$Total16_17[1], name = "1º Curso",
                                  sliced = TRUE),
                            list( y = Mat$Total16_17[2], name = "2º Curso"),
                            list( y = Mat$Total16_17[3], name = "3º Curso"),
                            list( y = Mat$Total16_17[4], name = "4º Curso")))

p2 = highchart() %>%
  hc_chart(type = "pie") %>%
  hc_title(text ="Porcentaje de alumnos por Curso en \n Estadística. Curso 16-17") %>%
  hc_xAxis(categories = Est$Curso) %>%
  hc_add_series(name = "% Alumnos",
                data = list(list( y = Est$Total16_17[1], name = "1º Curso",
                                  sliced = TRUE),
                            list( y = Est$Total16_17[2], name = "2º Curso"),
                            list( y = Est$Total16_17[3], name = "3º Curso"),
                            list( y = Est$Total16_17[4], name = "4º Curso")))

p3 = highchart() %>%
  hc_chart(type = "pie") %>%
  hc_title(
text ="Porcentaje de alumnos por Curso en \n Doble
Grado Física y Matemáticas. \n Curso 16-17") %>%
  hc_xAxis(categories = DobleFM$Curso) %>%
  hc_add_series(name = "% Alumnos",
```

```
                data = list(list( y = DobleFM$Total16_17[1], name = "1º Curso",
                                  sliced = TRUE),
                            list( y = DobleFM$Total16_17[2], name = "2º Curso"),
                            list( y = DobleFM$Total16_17[3], name = "3º Curso"),
                            list( y = DobleFM$Total16_17[4], name = "4º Curso")))

p4 = highchart() %>%
  hc_chart(type = "pie") %>%
  hc_title(
text ="Porcentaje de alumnos por Curso en \n Doble
Grado Matemáticas y Estadística.\n Curso 16-17") %>%
  hc_xAxis(categories = DobleME$Curso) %>%
  hc_add_series(name = "% Alumnos",
                data = list(list( y = DobleME$Total16_17[1], name = "1º Curso",
                                  sliced = TRUE),
                            list( y = DobleME$Total16_17[2], name = "2º Curso"),
                            list( y = DobleME$Total16_17[3], name = "3º Curso"),
                            list( y = DobleME$Total16_17[4], name = "4º Curso")))

bscols(p1,p2,p3,p4)
p = total_matriculados_ano_titulacion %>%
  gather(Curso, Matriculados,  -Titulacion) %>%
  mutate(Curso = substr(Curso,6,10 )) %>%
  arrange(Titulacion) %>%
  ggplot(aes(x = Curso, y = Matriculados, group = Titulacion, color = Titulacion)) +
  geom_line() +
  theme(legend.position="none")



 ggplotly(p)

aux = total_matriculados_ano_titulacion %>%
  gather(Curso, Matriculados,  -Titulacion) %>%
  mutate(Curso = substr(Curso,6,10 )) %>%
  arrange(Titulacion)

t1 = aux %>%
  filter(Titulacion == "Doble Grado en Física y Matemáticas") %>%
  select(Matriculados)
t1 = as.numeric(unlist(t1))
t1= ts(t1, start= 2013)
t1 = xts(t1, order.by = yearqtr(index(t1)))

t2 = aux %>%
  filter(Titulacion == "Doble Grado en Matemáticas y Estadística") %>%
  select(Matriculados)
t2 = as.numeric(unlist(t2))
t2 = xts(t2, order.by = yearqtr(index(t1)))

t4 = aux %>%
  filter(Titulacion == "Grado en Matemáticas") %>%
  select(Matriculados)
```

```
t4 = as.numeric(unlist(t4))
t4 = xts(t4, order.by = yearqtr(index(t1)))

t3 = aux %>%
  filter(Titulacion == "Grado en Estadística") %>%
  select(Matriculados)
t3 = as.numeric(unlist(t3))
t3 = xts(t3, order.by = yearqtr(index(t1)))

t5 = aux %>%
  filter(Titulacion == "Máster Universitario en Matemáticas") %>%
  select(Matriculados)
t5 = as.numeric(unlist(t5))
t5 = xts(t5, order.by = yearqtr(index(t1)))

matriculaciones <-cbind(t1,t2,t3,t4,t5)
colnames(matriculaciones) = c("DG F Y M", "DG M Y E", "E", "M", "M M")

dygraph(matriculaciones, main = "Evolucion Matriculados por titulacion") %>%
  dyRangeSelector() %>%
  dyAxis("x", label = "Curso Académico", valueRange = c(2013, 2017))

shinyAppFile(appFile = "G:/GRADO/TFG/CON DATOS FACULTAD/dygrap.R" )


texto = paste("Asignatura:",datos$Asignatura,"\n Alum15_16:", datos$Alum15_16,
              "\n Alum16_17:", datos$Alum16_17, "\n Titulacion:", datos$Titulacion)

scatterD3(data = as.data.frame(datos), x = Alum16_17, y = Alum15_16,
          col_var = Titulacion, tooltip_text = texto, caption =
            list(
title = "Comparación Matriculados en las diferentes asignaturas de cada titulación Curso
2016-2017 y 2015 -2016",
                          subtitle = "Definición Colores Titulación",
text =  " \n MATEMÁTICAS: Azul.     ESTADÍSTICA: Rojo.    MATEMÁTICAS Y ESTADÍSTICA: Naranja.
FÍSICA Y MATEMÁTICAS: Verde.   MÁSTER MATEMÁTICAS: Morado"), legend_width = 0)





texto = paste("Asignatura:",datos$Asignatura,"\n Alum14_15:", datos$Alum14_15,
              "\n Alum16_17:", datos$Alum16_17, "\n Titulacion:", datos$Titulacion)

scatterD3(data = as.data.frame(datos), x = Alum16_17, y = Alum14_15,
          col_var = Titulacion, tooltip_text = texto, caption =
list(title = "Comparación Matriculados en las diferentes asignaturas de
     cada titulación Curso 2016-2017 y 2014 -2015",
                          subtitle = "Definición Colores Titulación",
                          text =  " \n MATEMÁTICAS: Azul.     ESTADÍSTICA: Rojo.
     MATEMÁTICAS Y ESTADÍSTICA: Naranja.     FÍSICA Y MATEMÁTICAS: Verde.
     MÁSTER MATEMÁTICAS: Morado"), legend_width = 0)
```

```
p =ggplot(datos, aes(Alum16_17, repe)) +
  geom_point(aes(text =paste("Asignatura:", Asignatura), colour = Curso)) +
  facet_wrap(~ Titulacion)

ggplotly(p)

#Porcentaje de repetidores medios por curso en Matematicas

porcen_medio_repe = datos %>%
  group_by(Titulacion, Curso) %>%
  summarise(porcentaje_medio_repe = mean(tantorepe, na.rm = T))

rate = as.numeric(porcen_medio_repe %>%
  filter(Titulacion == "Grado en Matemáticas", Curso == 1) %>%
  ungroup() %>%
  select(porcentaje_medio_repe))

rate = round(rate, 2)

gauge(rate, min = 0, max = 100, symbol = '%',
      gaugeSectors(colors = c("lightgreen", "orange", "red"),
                   success = c(0, 10), warning = c(11, 20),
                   danger = c(21, 100)), abbreviateDecimals = 2,
      abbreviate = TRUE)

#Porcentaje de repetidores medios por curso en Matematicas

rate = as.numeric(porcen_medio_repe %>%
  filter(Titulacion == "Grado en Matemáticas", Curso == 2) %>%
  ungroup() %>%
  select(porcentaje_medio_repe))

rate = round(rate, 2)

gauge(rate, min = 0, max = 100, symbol = '%',
      gaugeSectors(colors = c("lightgreen", "orange", "red"),
                   success = c(0, 10), warning = c(11, 20),
                   danger = c(21, 100)), abbreviateDecimals = 2,
      abbreviate = TRUE)

#Porcentaje de repetidores medios por curso en Matematicas

rate = as.numeric(porcen_medio_repe %>%
  filter(Titulacion == "Grado en Matemáticas", Curso == 3) %>%
  ungroup() %>%
  select(porcentaje_medio_repe))

rate = round(rate, 2)

gauge(rate, min = 0, max = 100, symbol = '%',
      gaugeSectors(colors = c("lightgreen", "orange", "red"),
                   success = c(0, 10), warning = c(11, 20),
```

```
                    danger = c(21, 100)), abbreviateDecimals = 2,
      abbreviate = TRUE)

#Porcentaje de repetidores medios por curso en Matematicas

rate = as.numeric(porcen_medio_repe %>%
  filter(Titulacion == "Grado en Matemáticas", Curso == 4) %>%
  ungroup() %>%
  select(porcentaje_medio_repe))

rate = round(rate, 2)

gauge(rate, min = 0, max = 100, symbol = '%',
      gaugeSectors(colors = c("lightgreen", "orange", "red"),
                   success = c(0, 10), warning = c(11, 20),
                   danger = c(21, 100)), abbreviateDecimals = 2,
      abbreviate = TRUE)

#Porcentaje de repetidores medios por curso en Estadística

porcen_medio_repe = datos %>%
  group_by(Titulacion, Curso) %>%
  summarise(porcentaje_medio_repe = mean(tantorepe, na.rm = T))

rate = as.numeric(porcen_medio_repe %>%
  filter(Titulacion == "Grado en Estadística", Curso == 1) %>%
  ungroup() %>%
  select(porcentaje_medio_repe))

rate = round(rate, 2)

gauge(rate, min = 0, max = 100, symbol = '%',
      gaugeSectors(colors = c("lightgreen", "orange", "red"),
                   success = c(0, 10), warning = c(11, 20),
                   danger = c(21, 100)), abbreviateDecimals = 2,
      abbreviate = TRUE)

#Porcentaje de repetidores medios por curso en Estadística

rate = as.numeric(porcen_medio_repe %>%
  filter(Titulacion == "Grado en Estadística", Curso == 2) %>%
  ungroup() %>%
  select(porcentaje_medio_repe))

rate = round(rate, 2)

gauge(rate, min = 0, max = 100, symbol = '%',
      gaugeSectors(colors = c("lightgreen", "orange", "red"),
                   success = c(0, 10), warning = c(11, 20),
                   danger = c(21, 100)), abbreviateDecimals = 2,
      abbreviate = TRUE)

#Porcentaje de repetidores medios por curso en Estadística
```

```
rate = as.numeric(porcen_medio_repe %>%
  filter(Titulacion == "Grado en Estadística", Curso == 3) %>%
  ungroup() %>%
  select(porcentaje_medio_repe))

rate = round(rate, 2)

gauge(rate, min = 0, max = 100, symbol = '%',
      gaugeSectors(colors = c("lightgreen", "orange", "red"),
                   success = c(0, 10), warning = c(11, 20),
                   danger = c(21, 100)), abbreviateDecimals = 2,
      abbreviate = TRUE)

#Porcentaje de repetidores medios por curso en Estadística

rate = as.numeric(porcen_medio_repe %>%
  filter(Titulacion == "Grado en Estadística", Curso == 4) %>%
  ungroup() %>%
  select(porcentaje_medio_repe))

rate = round(rate, 2)

gauge(rate, min = 0, max = 100, symbol = '%',
      gaugeSectors(colors = c("lightgreen", "orange", "red"),
                   success = c(0, 10), warning = c(11, 20),
                   danger = c(21, 100)), abbreviateDecimals = 2,
      abbreviate = TRUE)

aux = datos %>%
  filter(Titulacion == "Grado en Matemáticas", Curso == 1)

highchart() %>%
  hc_chart(type = "bar") %>%
  hc_title(text = "Alumnos  Primer Curso Matematicas.\nComparación 16/17 - 15/16") %>%
  hc_xAxis(categories = aux$Asignatura) %>%
  hc_add_series(data = aux$Alum16_17,
                name = "16-17", color = "blue") %>%
  hc_add_series(data = aux$Alum15_16,
                name = "15-16", color = "orange")

aux = datos %>%
  filter(Titulacion == "Grado en Matemáticas", Curso == 2)

highchart() %>%
  hc_chart(type = "bar") %>%
  hc_title(text = "Alumnos  Segundo Curso Matematicas.\nComparación 16/17 - 15/16") %>%
  hc_xAxis(categories = aux$Asignatura) %>%
  hc_add_series(data = aux$Alum16_17,
                name = "16-17", color = "blue") %>%
  hc_add_series(data = aux$Alum15_16,
                name = "15-16", color = "orange")

aux = datos %>%
```

```
  filter(Titulacion == "Grado en Matemáticas", Curso == 3)

highchart() %>%
  hc_chart(type = "bar") %>%
  hc_title(text = "Alumnos Tercer Curso Matematicas.\nComparación 16/17 - 15/16") %>%
  hc_xAxis(categories = aux$Asignatura) %>%
  hc_add_series(data = aux$Alum16_17,
                name = "16-17", color = "blue") %>%
  hc_add_series(data = aux$Alum15_16,
                name = "15-16", color = "orange")

aux = datos %>%
  filter(Titulacion == "Grado en Matemáticas", Curso == 4)

highchart() %>%
  hc_chart(type = "bar", options3d = list(enabled = TRUE, beta = 15, alpha = 15)) %>%
  hc_title(text = "Alumnos  Cuarto Curso Matematicas.\nComparación 16/17 - 15/16") %>%
  hc_xAxis(categories = aux$Asignatura) %>%
  hc_add_series(data = aux$Alum16_17,
                name = "16-17", color = "blue") %>%
  hc_add_series(data = aux$Alum15_16,
                name = "15-16", color = "orange")

aux = datos %>%
  filter(Titulacion == "Grado en Estadística", Curso == 1)

highchart() %>%
  hc_chart(type = "bar") %>%
  hc_title(text = "Alumnos  Primer Curso Estadística.\nComparación 16/17 - 15/16") %>%
  hc_xAxis(categories = aux$Asignatura) %>%
  hc_add_series(data = aux$Alum16_17,
                name = "16-17", color = "blue") %>%
  hc_add_series(data = aux$Alum15_16,
                name = "15-16", color = "orange")

aux = datos %>%
  filter(Titulacion == "Grado en Estadística", Curso == 2)

highchart() %>%
  hc_chart(type = "bar") %>%
  hc_title(text = "Alumnos  Segundo Curso Estadística.\nComparación 16/17 - 15/16") %>%
  hc_xAxis(categories = aux$Asignatura) %>%
  hc_add_series(data = aux$Alum16_17,
                name = "16-17", color = "blue") %>%
  hc_add_series(data = aux$Alum15_16,
                name = "15-16", color = "orange")

aux = datos %>%
  filter(Titulacion == "Grado en Estadística", Curso == 3)

highchart() %>%
  hc_chart(type = "bar") %>%
  hc_title(text = "Alumnos Tercer Curso Estadística.\nComparación 16/17 - 15/16") %>%
```

```
  hc_xAxis(categories = aux$Asignatura) %>%
  hc_add_series(data = aux$Alum16_17,
                name = "16-17", color = "blue") %>%
  hc_add_series(data = aux$Alum15_16,
                name = "15-16", color = "orange")

aux = datos %>%
  filter(Titulacion == "Grado en Estadística", Curso == 4)

highchart() %>%
  hc_chart(type = "column", options3d = list(enabled = TRUE, beta = 15, alpha = 15)) %>%
  hc_title(text = "Alumnos  Cuarto Curso Estadística.\nComparación 16/17 - 15/16") %>%
  hc_xAxis(categories = aux$Asignatura) %>%
  hc_add_series(data = aux$Alum16_17,
                name = "16-17", color = "blue") %>%
  hc_add_series(data = aux$Alum15_16,
                name = "15-16", color = "orange") %>%
  hc_add_theme(hc_theme_google())

shinyAppFile(appFile = "G:/GRADO/TFG/CON DATOS FACULTAD/highchart.R" )
```

4.5.1. CÓDIGOS DE APLICACIONES SHINY

APLICACIÓN: Dygraph con Shiny

```
library(shiny)
library(dygraphs)


ui = shinyUI(fluidPage(

  titlePanel("Dygraph con Shiny"),

  sidebarLayout(
    sidebarPanel(

      selectInput("contra1", label = "Titulación 1",
                  choices = c("M", "E", "DG F Y M", "DG M Y E", "M M"),
                  selected = "M"),
      selectInput("contra2", label = "Titulación 2",
                  choices = c("M", "E", "DG F Y M", "DG M Y E", "M M"),
                  selected = "E"),
      checkboxInput("showgrid", label = "Show Grid", value = TRUE)
    ),
    mainPanel(
      dygraphOutput("dygraph")
    )
  )
))

server = shinyServer(function(input, output){

  output$dygraph <- renderDygraph({
```

```
t1 = c(0,105,285,466, 675)
t1 = ts(t1, start = 2013, frequency = 1)
t1 = xts(t1, order.by = yearqtr(index(t1)))
t2 = c(0,80,232, 400, 650)
t2 = ts(t2, start = 2013, frequency = 1)
t2 = xts(t2, order.by = yearqtr(index(t1)))
t3 = c(491,738, 793, 853, 869)
t3 = ts(t3, start = 2013, frequency = 1)
t3 = xts(t3, order.by = yearqtr(index(t1)))
t4 = c(3341,3802,3958,4100,4060)
t4 = ts(t4, start = 2013, frequency = 1)
t4 = xts(t4, order.by = yearqtr(index(t1)))
t5 = c(0,0,73,0,179)
t5 = ts(t5, start = 2013, frequency = 1)
t5 = xts(t5, order.by = yearqtr(index(t1)))

matriculaciones <-cbind(t1,t2,t3,t4,t5)
colnames(matriculaciones) = c("DG F Y M", "DG M Y E", "E", "M", "M M")

 pri1 = which(colnames(matriculaciones) == input$contra1)
 pri2 = which(colnames(matriculaciones) == input$contra2)

 matri = matriculaciones[,c(pri1, pri2)]

 name1 = switch(input$contra1,
        M = {print('Matemáticas')},
        E = {print('Estadística')},
        `DG F Y M` = {print('Física y Matemáticas')},
        `DG M Y E` = {print('Matemáticas y Estadística')},
        `M M` = {print('Máster Matemáticas')})

 name2 = switch(input$contra2,
                M = {print('Matemáticas')},
                E = {print('Estadística')},
                `DG F Y M` = {print('Física y Matemáticas')},
                `DG M Y E` = {print('Matemáticas y Estadística')},
                `M M` = {print('Máster Matemáticas')})

 dygraph(matri, main = "Comparación Número de Matriculados")  %>%
   dySeries(input$contra2, axis = 'y2') %>%
   dyAxis("y", label = paste("Matriculados",name1)) %>%
   dyAxis("y2", label = paste("Matriculados",name2)) %>%
   dyOptions(drawGrid = input$showgrid) %>%
   dyAnnotation("2013-1-1", text = "12-13",
                tooltip = "Curso 2012-2013", height = 20, width = 50) %>%
   dyAnnotation("2014-1-1", text = "13-14",
                tooltip = "Curso 2013-2014", height = 20, width = 50) %>%
   dyAnnotation("2015-1-1", text = "14-15",
                tooltip = "Curso 2014-2015", height = 20, width = 50) %>%
   dyAnnotation("2016-1-1", text = "15-16",
                tooltip = "Curso 2015-2016", height = 20, width = 50) %>%
   dyAnnotation("2017-1-1", text = "16-17",
                tooltip = "Curso 2016-2017", height = 20, width = 50)
```

```
   })

})

# Run the application
shinyApp(ui = ui, server = server)
```

APLICACIÓN: Comparación Alumnos por Asignatura

```
library(shiny)
library(highcharter)

load("G:/GRADO/TFG/Revisiones/170301_TFG_conmejoras_Salidas/TFG/datos.R")

ui = shinyUI(fluidPage(

  titlePanel("Comparación Alumnos por Asignatura"),

  helpText("Seleccione, para un curso y titulación, los años académicos a comparar"),

  sidebarLayout(
    sidebarPanel(

      selectInput("curso1", label = "Curso académico 1",
                  choices = c("2016-2017", "2015-2016", "2014-2015",
                              "2013-2014", "2012-2013"),
                  selected = "2016-2017"),

      selectInput("curso2", label = "Curso académico 2",
                  choices = c("2016-2017", "2015-2016", "2014-2015",
                              "2013-2014", "2012-2013"),
                  selected = "2015-2016"),

      selectInput("titu", label = "Titulación",
                  choices = c("Matemáticas", "Estadística"),
                  selected = "Matemáticas"),

      radioButtons("radio", label = "Curso",
                               choices = list("Primero" = 1, "Segundo" = 2,
                                              "Tercero" = 3, "Cuarto" = 4),selected = 1)

      ),
    mainPanel(
      highchartOutput("plot")
    )
  )
))

server = shinyServer(function(input, output){

  output$plot <- renderHighchart({

     curso =  input$radio
     titula = paste("Grado en", input$titu)
```

```
    aux = datos %>%
      filter(Titulacion == titula, Curso == curso)

    name1 = switch(input$curso1,
           `2016-2017` = {print('Alum16_17')},
           `2015-2016` = {print('Alum15_16')},
           `2014-2015` = {print('Alum14_15')},
           `2013-2014` = {print('Alum13_14')},
           `2012-2013` = {print('Alum12_13')}
           )

    name2 = switch(input$curso2,
                   `2016-2017` = {print('Alum16_17')},
                   `2015-2016` = {print('Alum15_16')},
                   `2014-2015` = {print('Alum14_15')},
                   `2013-2014` = {print('Alum13_14')},
                   `2012-2013` = {print('Alum12_13')}
    )

    titulo = paste("Alumnos de", input$radio, "de Grado", input$titu, ". \n",
                   "Comparación", input$curso1, "y", input$curso2)

    indi1 = which(colnames(aux) == name1)
    data1 = aux[[indi1]]

    indi2 = which(colnames(aux) == name2)
    data2 = aux[[indi2]]

    highchart() %>%
      hc_chart(type = "bar") %>%
      hc_title(text = titulo) %>%
      hc_xAxis(categories = aux$Asignatura) %>%
      hc_add_series(data = data1,
                    name = input$curso1, color = "blue") %>%
      hc_add_series(data = data2,
                    name = input$curso2, color = "orange")
  })

})

# Run the application
shinyApp(ui = ui, server = server)
```

Conclusiones

En este trabajo hemos estudiado como la inteligencia de negocios es actualmente una de las herramientas más utilizadas en el ámbito empresarial ya que proporciona información valiosa para la toma de decisiones.

Hemos visto también como el perfil de un profesional de la Estadística cubre algunas de las habilidades necesarias para el proceso de implantación y uso de la inteligencia de negocios. Los conocimientos adquiridos durante mi formación en el Grado en Estadística me han permitido comprender las diferentes fases de este tipo de aplicación: Bases de Datos, Ampliación de Inferencia Estadística, Estadística Computacional, Técnicas de Computación para la Estadística, Teoría de Decisión, etc.

La experiencia que he alcanzado en el uso del lenguaje R durante estos cuatro años me ha sido de gran utilidad para la creación de la aplicación web donde hemos situado el cuadro de mandos, esencial en la inteligencia de negocios.

A través de este trabajo he profundizado en el conocimiento de nuevos paquetes de R: Shiny, flexdashboard, leaflet, plotly, etc, que son muy utilizados actualmente. También, el haber estudiado el tema de la inteligencia de negocios y las aplicaciones comerciales más prestigiosas en este campo, los cuales no se estudian expresamente durante el Grado en Estadística, me han aportado una experiencia y visión nueva que creo puede ser de mucha utilidad en mi carrera profesional.

Como líneas futuras de trabajo espero seguir profundizando en el aprendizaje y uso de todos los componentes vinculados con la inteligencia de negocios, especialmente a través del uso avanzado del lenguaje R.

Bibliografía

"Diferencia entre Dato, Información y Conocimiento". URL http://iibi.unam.mx/~voutssasmt/documentos/dato%20informacion%20conocimiento.pdf.

Mª Carmen Ruiz Abellón. *Introducción a los Árboles de Decisión*, 2001. URL http://www.dmae.upct.es/~mcruiz/Telem06/Teoria/arbol_decision.pdf.

ajpdsof. *ajpdsof: Búsqueda de términos informático*. URL http://www.ajpdsoft.com/modules.php?name=Search.

JJ Allaire. *flexdashboard: R Markdown Format for Flexible Dashboards*, 2017. URL https://CRAN.R-project.org/package=flexdashboard. R package version 0.4.

Tomas Aluja. La minería de datos, entre la estadística y la inteligencia artificial. *Universitat Polit'ecnica de Catalunya*, 25(3):479–498, 2001.

amsantac: Soluciones eficientes para el procesamiento de datos geoespaciales. *Mapeo web con Leaflet y R*, 2015. URL http://amsantac.co/blog/es/r/2015/08/11/leaflet-R-es.html.

ANER. *Cuadros de Mando con QlikView*. URL http://www.aner.com/business-intelligence-qlikview.html.

Lenin Alevski Arias Huerta. *Inteligencia de Negocios*, 2017. URL https://es.slideshare.net/Alevsk/inteligencia-de-negocios-9667502?next_slideshow=1.

Adam Aspin. *Business Intelligence with SQL Server Reporting Services*. Apress, 2015.

Font Awesome. *The complete set of 675 icons in Font Awesome 4.7.0*. URL http://fontawesome.io/icons/.

Bernard Pavel Barreto Véliz. *Inteligencia de Negocios*. URL http://www.monografias.com/trabajos75/inteligencia-negocios/inteligencia-negocios2.shtml.

Edgar David Padierna Bedoya. *Pentaho - Business Intelligence*, 2012. URL https://prezi.com/zgx29l0rwmhh/pentaho-business-intelligence/.

beta Rstudio. *Example ggplotly geoms*. URL https://beta.rstudioconnect.com/jjallaire/htmlwidgets-ggplotly-geoms/htmlwidgets-ggplotly-geoms.html#geom_density.

BetterBuys. *The Definitive Guide to Business Intelligence*, a. URL https://www.betterbuys.com/bi/definitive-guide-bi/.

Elizabeth Mazenko BetterBuys. *Microsoft Power BI Review*, b. URL https://www.betterbuys.com/bi/reviews/microsoft-power-bi/.

Elizabeth Mazenko BetterBuys. *Qlik Sense Review*, c. URL https://www.betterbuys.com/bi/reviews/qlik-sense/.

Elizabeth Mazenko BetterBuys. *Tableau Software Review*, d. URL https://www.betterbuys.com/bi/reviews/tableau-business-intelligence/.

Justin Heinze BetterBuys. *Qlikview Review*, e. URL https://www.betterbuys.com/bi/reviews/qlikview-business-intelligence/.

Justin Heinze BetterBuys. *Pentaho Business Intelligence Review*, f. URL https://www.betterbuys.com/bi/reviews/pentaho-business-intelligence/.

RStudio Blog. *Flexdashboard: Easy interactive dashboards for R*, 2016. URL https://blog.rstudio.org/2016/05/17/flexdashboard-easy-interactive-dashboards-for-r/.

Bootstrap. *The most popular HTML, CSS, and JS framework for developing responsive, mobile first projects on the web*. URL https://getbootstrap.com/components/.

Information Builders. *"Business Intelligence"*. URL http://www.informationbuilders.com/business-intelligence.

Luis Alfonso Camarero Rioja. *Estadísitca para la investigación social*. Garceta Grupo Editorial, 2010.

Damián Jorge Matich Carlos Alberto Ruiz, Marta Susana Basualdo. *Redes Neuronales: Conceptos Básicos y Aplicaciones*. Universidad Tecnológica Nacional – Facultad Regional Rosario. Departamento de Ingeniería Química, 2001.

Jon Kepa; Alzua-Sorzabal Aurkene Castánder Martín, Iñigo; Gerrikagoitia Arren. *Estudio comparativo de las herramientas de Business Intelligence: Empoderando el criterio de selección a las PYMEs*, 2013. URL https://www.researchgate.net/publication/259442931_Estudio_comparativo_de_las_herramientas_de_Business_Intelligence_Empoderando_el_criterio_de_seleccion_a_las_PYMEs.

Daniel Satín González Cesar Pérez López. *Minería de datos: técnicas y herramientas*. THOMSON, 2008.

Winston Chang, Joe Cheng, JJ Allaire, Yihui Xie, and Jonathan McPherson. *shiny: Web Application Framework for R*, 2016. URL https://CRAN.R-project.org/package=shiny. R package version 0.14.2.

Diego Morales Cifuentes. *Trabajo Fin de Grado: Minería de datos. Aplicaciones de técnicas predictivas.* Dep. de Organización Industrial y Gestión de Empresas I Escuela Técnica Superior de Ingeniería Universidad de Sevilla, 2016.

Debra Dalgleish. *Excel Pivot Tables Recipe Book*. Apress, 2006.

knowledge is the goal DataPrix. *Cubos OLAP (On-Line Analytic Processing)*, 2010. URL http://www.dataprix.com/blogs/respinosamilla/cubos-olap-line-analytic-processing.

Knowledge is the goal. Carlo Placani DataPrix. *QlikView, simplificando el análisis para todos*, 2009. URL http://www.dataprix.com/es/qlikview-simplificando-el-lisis-para-todos.

Departamento de Estadística e I.O. Universidad de Granada. *Análisis Cluster*. URL http://www.ugr.es/~bioestad/_private/cpfund7.pdf.

Universidad de Granada: Fernando Berzal. *Reglas de asociación*. URL http://elvex.ugr.es/decsai/intelligent/slides/dm/D2%20Association.pdf.

Martha De León. *Fundamentos de la inteligencia de negocios: bases de datos y administración de la información*, 2015. URL https://prezi.com/mth-dgnuuvga/fundamentos-de-la-inteligencia-de-negocios-bases-de-datos-y/.

María de los Ángeles Ibarra. *Procesamiento Análitico en Linea (OLAP)*. Universidad Nacional del Nordeste. Facultad de Ciencias Exactas, Naturales y Agrimensura, 2006.

Inteligencia de Negocio.mx. *Definiendo los objetivos de un proyecto de Business Intelligence*, 2011. URL http://inteligenciadenegocio.mx/blog/definiendo-los-objetivos-de-un-proyecto-de-business-intelligence.

Trabajos de sistemas. *Fundamentos de la Inteligencia de Negocios: Base de Datos y Gestion de la Información*, 2015. URL https://trabajosdesistemas.wordpress.com/2015/01/06/fundamentos-de-la-inteligencia-de-negocios-base-de-datos-y-gestion-de-la-unformacion/.

definicionabc. *Definición de Datawarehouse*. URL http://www.definicionabc.com/tecnologia/datawarehouse.php.

Josep Curto Díaz. *Introducción al Business Intelligence*. UOC, 2010.

Erika Vilches González e Iván A. Escobar Broitman. *Minería de Datos*, 2009. URL http://erika--vilches.azurewebsites.net/wp-content/uploads/2009/10/mineria_datos.pdf.

Optimiza la eficiencia de tu empresa El blog de WorkMeter. *Principales herramientas de Business Intelligence*, 2012. URL http://es.workmeter.com/blog/bid/192978/Principales-herramientas-de-Business-Intelligence.

Arcplan Enterprise. *Common Charts*, 2012. URL http://www.arcplan.com/index.php?id=153.

Qué es WebSocket. *Programación reactiva*, 2013. URL http://queeswebsocket.blogspot.com.es/2013/01/que-es-websocket.html.

Consulting Evolve it and Solutions. *Las herramientas básicas de un Business Intelligence Software*, 2016. URL http://www.evolve-it.com.mx/las-herramientas-basicas-de-un-business-intelligence-software/.

Klipfolio: Dashboard Examples. *ER Status*. URL https://www.klipfolio.com/resources/dashboard-examples/healthcare/er-status.

Expandiabusiness. *Tableau-software*. URL https://es.slideshare.net/Expandiabusiness/presentacin-tableau-software.

Forrester. *"Business Intelligence"*. URL https://www.forrester.com/Business-Intelligence.

Business Intelligence fácil. *Cuadrante mágico Business Intelligence y Business Analytics*, 2016. URL https://www.businessintelligence.info/mercado/gartner-bi-2016.html.

Ileana Garcia. *Tableau*, 2015. URL https://prezi.com/ljbrudqhcjkg/tableau/.

Gartner. *Critical Capabilities for Business Intelligence and Analytics Platforms*, 2016. URL https://www.gartner.com/doc/reprints?id=1-317E15Q&ct=160316&st=sb.

Miguel Gaspar. *Learning Pentaho CTools*. PACKT publishing, 2016.

Jay Gendron. *Introduction to R for Business Intelligence*. PACKT publishing, 2016.

Gotrecillo: Gitbooks. *Programación reactiva*. URL https://gotrecillo.gitbooks.io/meteordocumentation/content/programacion_reactiva.html.

GitHub. *Introduction To R for Business Intelligence*. URL https://github.com/PacktPublishing/Introduction-To-R-For-Business-Intelligence.

Rstudio Github. *dygraphs for R*, a. URL http://rstudio.github.io/dygraphs/.

Rstudio Github. *DT: An R interface to the DataTables library*, b. URL https://rstudio.github.io/DT/.

Rstudio github. *Crosstalk*. URL https://rstudio.github.io/crosstalk/.

Aníbal Goicochea. *Tipos de gráficos según el problema a representar*, 2012. URL https://anibalgoicochea.com/2012/12/31/tipos-de-graficos-segun-el-problema-a-representar/.

Rodríguez José Ramón Guitart Isabel González Farran, Xavier. *¿Cómo planificar un proyecto de inteligencia de negocio?* UOC, 2016.

Información sin límites Gravitar. *¿Qué es pentaho?* URL http://gravitar.biz/pentaho/.

solución de gestión para PYMes Grupo IGN. *Comparativa entre las principales herramientas Business Intelligence*, 2016. URL http://ignsl.es/principales-herramientas-business-intelligence/.

Guidobeymarperezlopez. *Problemas con el entorno de archivos tradicionales*, 2015. URL https://guidobeymarperezlopez.wordpress.com/2015/01/06/problemas-con-el-entorno-de-archivos-tradicionales/.

Jan Górecki. *R in Business Intelligence*, 2015. URL https://www.r-bloggers.com/r-in-business-intelligence/.

Hipertextual. *Markdown: qué es y cómo aprovecharlo*, 2013. URL https://hipertextual.com/archivo/2013/04/que-es-markdown/.

htmlwidgets for R. *Bring the best of JavaScript data visualization to*. URL http://www.htmlwidgets.org/.

IBM. *"La Importancia de la Inteligencia de Negocios Aplicada a Empresas Medianas"*. URL http://www.ibm.com/developerworks/ssa/local/data/dm-bi-pymes/.

Ionicons. *The premium icon font for Ionic Framework*. URL http://ionicons.com/.

Roberto Bosco J. *PowerPivot for Advanced Reporting and Dashboards*. PACKT publishing, 2013.

Arshad Khan. *Jumpstart Tableau. A step-by-step guide to better data visualization*. Apress, 2016.

Len Kiefer. *State-to-state migration in 2014*. URL https://public.tableau.com/profile/leonard.kiefer#!/vizhome/State-to-statemigrationin2014/Mobilitygraphic.

Len Kiefer. *Converting a Tableau dashboard to a Flexdashboard*, 2017. URL http://lenkiefer.com/2017/01/26/convert-tableau-to-r.

B. W. Lewis. *Globe example, threejs package*, a. URL http://bwlewis.github.io/rthreejs/bling/index.html.

B. W. Lewis. *The threejs package: three.js widgets for R*, b. URL http://bwlewis.github.io/rthreejs/.

Business Logicalis and technology working as one. *Todo sobre los Cubos OLAP y estructuras multidimensionales*, 2015. URL https://blog.es.logicalis.com/analytics/cubos-olap-y-estructuras-multidimensionales-todo-lo-que-hay-que-saber.

Computer World. Sharon Machlis. *Great R packages for data import, wrangling visualization*, 2017. URL http://www.computerworld.com/article/2921176/business-intelligence/great-r-packages-for-data-import-wrangling-visualization.html.

Ennio Prada Madrid. Los insumos invisibles de decisión: Datos, información y conocimiento. *Pontificia Universidad Javeriana.*, 14(11):183–196, 2008.

MangoSolutions. *ValidR*. URL http://www.mango-solutions.com/wp/products-services/products/validr/.

Jose Manuel Martínez. *Cuadros de Mandos Integrales (CMI), mirando desde arriba*, 2011. URL https://www.clavei.es/blog/2011/cuadros-de-mandos-integrales-cmi-mirando-desde-arriba/.

Mundo ERP. Sergio Martínez. *Funcionamiento de un entorno de Business Intelligence (BI)*. URL http://mundoerp.com/blog/funcionamiento-entorno-business-intelligence-bi/.

Online Media Masters. *7 Google Analytics Custom Dashboard Examples To Segment SEO, Social Media, Mobile, And Other Metrics*, 2017. URL http://www.onlinemediamasters.com/google-analytics-custom-dashboard-examples/.

Maximizer. *"7 reasons why bussines intelligence is vital to business success"*. URL https://www.maximizer.com/blog/7-reasons-why-business-intelligence-is-vital-to-business-success/.

mcpasin Shinyapss. *Playing with Google Analytics Data*. URL https://mcpasin.shinyapps.io/PlayingGoogleAnalyticsDataViz/.

Show me Shiny. *Gallery of R web APPS*. URL https://www.showmeshiny.com/.

Developer Network Microsoft. *Conceptos básicos de ODBC*, a. URL https://msdn.microsoft.com/es-es/library/thzzea08.aspx.

Power BI Microsoft. *Power BI Microsoft*, b. URL https://powerbi.microsoft.com/es-es/.

Monografías. *Inteligencia de Negocios (BI)*. URL http://www.monografias.com/trabajos14/bi/bi.shtml.

mrc's Cuo of Joe BLog. *7 critical elements of effective dashboards*, 2014. URL http://www.mrc-productivity.com/blog/2014/10/7-critical-elements-of-effective-dashboards/.

CIO: Ryan Mulcany. *Business Intelligence Definition and Solutions*, 2007. URL http://www.cio.com/article/2439504/business-intelligence/business-intelligence-business-intelligence-definition-and-solutions.html.

Gustavo Nipas. *10 Sistemas gestores de base de datos*. URL https://es.slideshare.net/nipas/10-sgbd.

Ocalab. *Preocúpate por diseñar un buen dashboard y obtendrás un gran reporting*, 2014. URL http://ocalab.es/preocupate-por-disenar-un-buen-dashboard-y-obtendras-un-gran-reporting/.

OLAP. *"What is Business Intelligence (BI)?"*. URL http://olap.com/learn-bi-olap/olap-bi-definitions/business-intelligence/.

Jordi Ollé. *Los 7 pasos para convertirte en un data scientist en R. Un camino de aprendizaje ordenado y sin agobios*, 2016. URL http://conceptosclaros.com/los-7-pasos-para-convertirte-en-data-scientist-en-r/#Ploteargraficos_interactivos.

Oracle. *Paper Oracle: Qué es inteligencia de negocios*. URL http://www.oracle.com/ocom/groups/public/@otn/documents/webcontent/317529_esa.pdf.

Dejan Sarka Dustin Ryan Patrick LeBlanc, Jessica M.Moss. *Applied Microsoft Business Intelligence*. Wiley, 2015.

PCmag. *Business Intelligence: How it Works*, 2001. URL http://www.pcmag.com/article2/0,2817,7435,00.asp.

Pentaho. *Pentaho*, a. URL http://www.pentaho.com.

Pentaho. *Dashboard: Chicago 10 years of crime*, b. URL http://ww3.pentaho.com/pentaho/plugin/chicagoCrime/api/main?userid=pentaho&password=demo.

Pentaho. *Dashboard: Theme Park*, c. URL http://ww3.pentaho.com/pentaho/api/repos/:public:PublicDemo:ThemePark:home.wcdf/generatedContent?userid=pentaho&password=demo.

Finanzas Personales. *Finanzas Personales: "Tres Claves para lograr el éxito en su empresa"*. URL http://www.finanzaspersonales.com.co/trabajo-y-educacion/articulo/tres-claves-para-lograr-exito-su-empresa/48612.

Carlos Pesquera. *Enfoque y Posicionamiento entre Qlikview vs. Qliksense*, 2015. URL http://carlospesquera.com/enfoque-posicionamiento-qlikview-vs-qliksense/.

Plotly. *Plotly R Library*. URL https://plot.ly/r/.

PowerData. *Tipos y función de los gestores de bases de datos*, 2015. URL http://blog.powerdata.es/el-valor-de-la-gestion-de-datos/bid/406547/tipos-y-funci-n-de-los-gestores-de-bases-de-datos.

Qlik. *Qlik*. URL http://www.qlik.com/es-es/.

Mundo Qlik. *QlikView vs Qlik Sense*, 2015. URL http://mundoqlik.com/qlikview-vs-qlik-sense.

R-Bloggers. *htmlwidgets: JavaScript data visualization for R*, 2014. URL https://www.r-bloggers.com/htmlwidgets-javascript-data-visualization-for-r/.

R-Bloggers. *Interactive time series with dygraph*, 2015. URL https://www.r-bloggers.com/interactive-time-series-with-dygraphs/.

ReBolicand-data. *Shiny: Crear mini aplicaciones con R-Studio*, 2016. URL http://rebolicando.blogspot.com.es/2016/01/shiny-crear-mini-aplicaciones-con-r.html.

Stephen Redmond. *Mastering QLikView*. PACKT publishing, 2014.

Santamaría Fernando Reporte Digital. *Visualización de datos: una estrategia empresarial para la inteligencia del negocio*, 2014. URL http://reportedigital.com/iot/visualizacion-datos-estrategia-empresarial-inteligencia-negocio/.

Youtube: Carlos Reyes. *Mi primer QlikView desde Excel*. URL https://www.youtube.com/watch?v=NTUe6bj8J7g.

Rstudio: Rmarkdown. *Flexdashboard: Easy interactive dashboards for R*. URL http://rmarkdown.rstudio.com/flexdashboard/index.html.

Sebastián Rodríguez Robotham. *DataWarehouse e inteligencia de Negocios*, 2012. URL https://es.slideshare.net/sebasrod/introduccin-al-datawarehouse-e-inteligencia-de-negocios.

María Carina Roldán. *Pentaho Data Integration, Beginner's Guide. Second Edition*. PACKT publishing, 2013.

Rpubs. *Example heatmap*. URL http://rpubs.com/jcheng/mtcars-heatmap.

Rstudio. *Hoja de referencia R Markdown*, a. URL https://www.rstudio.com/wp-content/uploads/2015/03/rmarkdown-spanish.pdf.

Rstudio. *Hoja de referencia Shiny*, b. URL https://www.rstudio.com/wp-content/uploads/2015/03/shiny-spanish.pdf.

RStudio. *Crosstalk: RStudio Github*, 2017. URL https://rstudio.github.io/crosstalk/.

Github RStudio. *Shiny Dashboard*. URL http://rstudio.github.io/shinydashboard/structure.html.

DataCamp: Jeffrey Ryan. *Manipulating Time Series Data in R with xts zoo*. URL https://www.datacamp.com/courses/manipulating-time-series-data-in-r-with-xts-zoo.

Stochastic Coder. Jonathan Scholtes. *Starting with Rmarkdown Flexdashboard*, 2016. URL https://stochasticcoder.com/2016/06/22/starting-with-r-markdown-flex-dashboard/.

G.R. Serrano. *Una aplicación web con Shiny*, 2013. URL http://www.grserrano.net/wp/2013/03/una-aplicacion-web-con-shiny/l.

Julio M. Sevilla. *Aplicaciones web con Shiny*, 2015. URL http://www.epidom.es/blog/2015/01/aplicaciones-web-con-shiny/.

Julio M. Sevilla Sánchez. *Modelo de estimación del volumen en función de la edad en masas clonales de Eucalyptus globulus Labill. en la provincia de Huelva*. URL https://juliomsevilla.shinyapps.io/ModeloGlobulusHuelva/.

shiny by Rstudio. *Teach yourself Shiny*. URL https://shiny.rstudio.com/tutorial/.

sinnexus. *¿Qué es Business Intelligence?* URL http://www.sinnexus.com/business_intelligence/index.aspx.

Socius. *"Why do I need a business intelligence solution?"*. URL http://www.socius1.com/why-do-i-need-a-business-intelligence-solution/.

Youtube: Quabu Solutions. *Ejemplo cuadro de mandos Qlik Sense - KPI's de dirección*. URL https://www.youtube.com/watch?v=HEoWTvSY7Mc.

Tableau. *Tableau*, a. URL https://www.tableau.com/es-es/.

Tableau. *Software de dashboards que hace un seguimiento de los problemas antes de que se conviertan en dolores de cabeza*, b. URL https://www.tableau.com/es-es/stories/workbook/dashboard-software-tracks-issues-they-become-headaches.

Techopedia. *"Definition - What does Business Intelligence (BI) mean?"*. URL https://www.techopedia.com/definition/345/business-intelligence-bi.

SearchDataCenter en Español. Cristián Vera-Cruz TechTarget. *Inteligencia de negocios: el secreto está en los datos*. URL http://searchdatacenter.techtarget.com/es/cronica/Inteligencia-de-negocios-el-secreto-esta-en-los-datos.

Antonio Universidad de Murcia: González, Aurora y Maurandi. *Shiny. Entornos web con R*. URL http://www.um.es/ae/tShiny/TallerShiny-reducido-web.html.

Manuel Universidad de Vigo: Oviedo de la Fuente. *Shiny: crear una aplicación web interactiva desde R*, 2013. URL http://eio.usc.es/pub/moviedo/descargas/Shiny.html#31.

Ramnath Vaidyanathan, Yihui Xie, JJ Allaire, Joe Cheng, and Kenton Russell. *htmlwidgets: HTML Widgets for R*, 2016. URL https://CRAN.R-project.org/package=htmlwidgets. R package version 0.8.

VhcRodriguez. *Empresas que trabajan con inteligencia de negocios*. URL https://es.slideshare.net/VhcRodriguez/empresas-que-trabajan-con-inteligencia-de-negocios-mercadotecnia-electronica.

vieraroman.blogspot. *Organización De Datos En Un Entorno Tradicional De Archivos*. URL http://vieraroman.blogspot.com.es/p/71-organizacion-de-datos-en-un-entorno.html.

Hadley Wickham. *stringr: Simple, Consistent Wrappers for Common String Operations*, 2016. URL https://CRAN.R-project.org/package=stringr. R package version 1.1.0.

Wikipedia. *Wikipedia: Ïnteligencia Empresarial"*, a. URL https://es.wikipedia.org/wiki/Inteligencia_empresarial.

Wikipedia. *Calidad de datos*, b. URL https://es.wikipedia.org/wiki/Calidad_de_datos.

Wikipedia. *Java DataBase Connectivity*, c. URL https://es.wikipedia.org/wiki/Java_Database_Connectivity.

Wikipedia. *Procesamiento por lotes*, d. URL https://es.wikipedia.org/wiki/Procesamiento_por_lotes.

CTR World. *Tableau – Pros and Cons*. URL http://www.ctrworld.com/tableau-pros-and-cons/.

wwwhat's new. *Qué es y cómo funciona Google Analytics*, 2013. URL http://wwwhatsnew.com/2013/08/27/que-es-y-como-funciona-google-analytics/.

Yihui Xie. *Dynamic Documents with R and knitr*. Chapman and Hall/CRC, Boca Raton, Florida, 2nd edition, 2015. URL http://yihui.name/knitr/. ISBN 978-1498716963.

Yihui Xie. *bookdown: Authoring Books and Technical Documents with R Markdown*, 2016a. URL https://github.com/rstudio/bookdown. R package version 0.3.2.

Yihui Xie. *knitr: A General-Purpose Package for Dynamic Report Generation in R*, 2016b. URL https://CRAN.R-project.org/package=knitr. R package version 1.15.1.

TechonologyAdvice: Youtube. *Guide to Business Intelligence. Business Intelligence Tutorial*, 2014. URL https://www.youtube.com/watch?v=jkCCnwvO_fg.

bring your data to life Yurbi. *Straight Talk Review: Strengths and Weaknesses of Microsoft Power BI*, 2016. URL https://www.yurbi.com/blog/straight-talk-review-strengths-and-weaknesses-of-microsoft-power-bi/.